LJ-81-13C

NATIONAL MARINE FISHERIES SERVICE
SOUTHWEST FISHERIES CENTER
P.O. BOX 271
LA JOLLA, CA 92038

SOUTHWEST FISHERIES CENTER

May 1981

Development of a Numerical Simulation of Purse Seine Fishing

I. Numerical Simulation of Net Fishing

II. Extension of the Numerical Simulation of the Purse Seine Fishing Process

BY

T. N. Delmer

AND

T. C. Stephens

Library
NOAA, National Marine Fisheries Service
SWFC Honolulu Laboratory F/SWC2
2570 Dole Street
Honolulu, HI 96822-2396

ADMINISTRATIVE REPORT No. LJ-81-13C

DEVELOPMENT OF A NUMERICAL SIMULATION OF PURSE SEINE FISHING

I. Numerical Simulation of Net Fishing[1]

II. Extension of the Numerical Simulation of the Purse Seine Fishing Process[2]

by

T. N. Delmer and T. C. Stephens

Beers Associates
P.O. Box 2549
Reston, Virginia 22090

May 1981

This report was prepared by T. N. Delmer and T. C. Stephens to the Science Applications, Inc. under contract number NASO 78-0266[1] and to Beers Associates, Inc. under contract number 80-ABC-00182[2] for the National Marine Fisheries Service, Southwest Fisheries Center, La Jolla, California. The statements, findings, conclusions and recommendations herein are those of the author(s) and do not necessarily reflect the views of the National Marine Fisheries Service. James M. Coe of the Southwest Fisheries Center served as Chief Official Technical Representative.

DEVELOPMENT OF A NUMERICAL SIMULATION OF PURSE SEINE FISHING

I. Numerical Simulation of Net Fishing

PREFACE

The American Tuna Industry is highly productive and extremely valuable. As it is a product rather than a service industry, it plays an important role in the nation's economy.

The tuna seiners, representing the vast majority of the industry, are required to reduce the porpoise kill that is incidental to their fishing technique. It is imperative that this reduction be made with minimal detriment to the efficiency of the industry. Impressive reduction has already been made by minor, but ingenious, modifications of fishing gear and techniques. Still, further reduction in the porpoise kill is required.

The main physical process that determines the porpoise kill is the net behavior. Consequently, understanding and being able to predict this behavior and the influence of modifications in the gear and procedures, are important in reducing the kill. These same factors are also those which determine and can increase the efficiency of the net in catching tuna.

There are two principal methods of determining the net behavior: experimental and analytical. The experimental method has the overwhelming advantage that the results are real and irrefutable. The disadvantages of the method are numerous: the expense of conducting experiments is great; ocean parameters cannot be controlled and are difficult to measure; and changing net configurations requires a great deal of work and money. In direct opposition to this, the analytical method has the disadvantage that the results are not real and may be challenged. This disadvantage can be eliminated by comparing the results of analysis with some experimental measurements. If the analysis is verified by this comparison, the advantages of the second method are realized: low cost, and control of ocean and net parameters. The key to obtaining these advantages is the experimental verification.

A program was initiated by the Southwest Fisheries Center to develop a method of computer simulation of net behavior in the seining process, since there is reason to believe that the analysis would agree with experiment. The reason for this optimism, in face of the fact that real net behavior is very complex, is that the complexity is quantitative rather than qualitative. The action of each part of the net is governed primarily by the parameters of local tension and water drag; thus, the basic interaction is qualitatively relatively uncomplicated. However, because these parameters vary widely over the large physical size of the net, the overall net behavior is complex. The digital computer is ideally suited to handle such quantitatively complex problems.

ABSTRACT

The equations describing fishing with a net and boat are formulated for computer solution. The net is described by a large macromesh that results from a numerical formulation of the partial differential equation for the net using the method of lines. The problem is reduced to solving a set of ordinary differential equations. A standard computer code, GEARS, is used to integrate the equations. A user oriented code is used to construct a net to initialize the simulation. The code can model nets, cables constrained to move with the net by rings, free cables, and the boat at low speed. However, the simulation cannot at present describe problems with radical changes in topological properties.

Digitized by the Internet Archive
in 2025

https://archive.org/details/administrative-report-lj_1981-05_lj-81-13c

TABLE OF CONTENTS

Section		Page
1.	INTRODUCTION	1
2.	GEAR AND PROCEDURE	5
3.	MATHEMATICAL FORMULATION OF THE PROBLEM	14
	3.1 Equations Describing Net Material	14
	3.2 Equations Describing the Cables	25
	3.3 Equations Describing the Seiner	31
4.	MODELING PROCEDURE AND NUMERICAL METHODS	39
	4.1 Numerical Model	39
	4.2 Accuracy of the Model	40
	4.3 Selection of Numerical Methods	41
	4.4 Potential Problems	42
	4.5 General Comments	43
5.	NUMERICAL MODEL	45
	5.1 Numerical Model of the Net	45
	5.2 Numerical Model of Cables	46
	5.3 Numerical Formulation of the Equation of Motion of a Constrained Cable	47
	5.3.1 Free Hanging Cable	51
	5.3.2 Constrained Cable	53
6.	SUMMARY	61
	ACKNOWLEDGEMENTS	63
	BIBLIOGRAPHY RELATING TO NET MODELING	65
	BIBLIOGRAPHY RELATING TO MATHEMATICAL METHODS	71
	BIBLIOGRAPHY RELATING TO CABLE METHODS	75
	APPENDIX	77

LIST OF FIGURES

Figure		Page
1.	Composite View: Setting the Net and Pursing . .	6
2.	Composite View: Rings up and Rolling the Net . .	7
3.	Purse Cable, Rings and Lead Line	8
4.	Cork Line, Bunch Lines	9
5.	Net Membrane Macro-Mesh Similarity	15
6.	Subunit of Net	17
7.	Drag and Lift Coefficients	22
8.	Multiplier for Drag Coefficient to Correct for Use as Drag Coefficient in Nets	23
9.	Schematic Diagram of Purse Cable	49
10.	Schematic Diagram of Free Hanging Cable	50

LIST OF TABLES

Table		Page
1.	Phases of the Process that Must be Considered as Seperate Problems	12
2.	Special Features in Different Phases and Transition Characteristics	13
3.	Coordinates for Seiner	33
4.	Tension Forces	33
5.	Propulsion Forces	34
6.	Components of Force and Torque	34
7.	Coefficients a and b for Water Drag on Seiner	36
8.	Equations of Motion of the Seiner	37
9.	Error When Using the Method of Lines	41
10.	Types of Constrained Cables	48
11.	Equations Describing Free Hanging Cable	54
12.	Equations Describing the Constrained Cable	58
13.	Seiner Parameters	78

1. INTRODUCTION

This report contains the result of an effort to develop a computer code to simulate the purse-seining process. This effort was in two parts: Phase I that partially formulated the problem and Phase II that expanded the results of the first phase and generated the computer code. The code is to allow the Southwest Fisheries to design procedures and nets that will increase the tuna catch and decrease porpoise kill. The code is incomplete in that certain important effects although formulated, have not yet been interfaced and it has not been exercised to a sufficient extent to give confidence in its use. These limitations will be described individually and summarized in the conclusion.

There are two types of audience that will be interested in this report; the biological scientist and the physical scientist. Most of those scientists concerned with the tuna industry are biologists and have an intimate knowledge of the purse-seining process but limited background in the mathematics required here. Conversely, those who have the mathematical background do not have experience with seining for tuna. The biologists must remember that understanding the mathematics requires specialized training and the mathematician must realize that the purse seining process is one that can only be understood by repeatedly observing the process of catching tuna.

Construction of the simulation proceeds by observing the seining process and equipment, breaking the phenomena down into various time and equipment components, modeling the various pieces of equipment and then recombining the pieces in the proper time history. The salient features of seining are described in section 2. This might also serve as a rough background for those unfamiliar with the process.

There has been little computational work on nets in the past. The reason for this is that the Germans, Japanese, and Russians are those most interested in analysis of nets but they do

not have the access to computers such as exists in the United States. Examples of the numerical calculations done by these workers are found in Ivanov (1974), Koritzkz (1977), Blinov (1973), Leizke (1976), Crewe (1973), Matuda and Sannomiya (1976). The problems that were addressed were static problems, either trawls or drift nets, with the net material being uniform. The methods do not apply to the purse seining process which is, from the beginning, a thoroughly dynamic procedure.

In the English literature, the problem of cabled structures in air, water, or concrete is studied to some extent with the use of computer techniques. The problems deal with antennas, moored objects and structures. Some examples are Sangster and Batchelor (1979), Felippa (1974), and Morris and Birnstiel (1974). These authors deal with problems that are simplified either because of their size or because the structure is uniform or static.

In contrast to previous work, simulating the seining process requires the solution of a large time dependent, inhomogeneous problem. Considering that up to eight hundred knots (each with six degrees of freedom) may be required to describe the net, the problem is within a factor of two of the estimated state of the art in structural problems (Felippa and Clark, 1978, page 4-2 Figure 7, Hotz, 1979). There are also many phenomena that must be simulated here that do not occur in other problems. Most notable of these are the cables that slip through rings attached to the net with the cable length changing orders of magnitude from one time to another in the problem. In fact, all the cables must be simulated as running cables in contrast to the methods used in the literature. The scaling required by the net is unique. The hundreds of millions of knots and bars in a net cannot be modeled and must be scaled in some fashion rather than modeling every bar as can be done in some structures.

However, it is no surprise that the purse seining process requires unique simulation techniques; the process is itself unique. The simulation proceeds in the following fashion:

- Definition of process in generic detail
- Derivation of equations governing the process
- Determination of numerical methods to integrate the equations
- Formulation of the numerical version of the equations to be integrated
- Development of the computer code to allow construction of the net and interfacing of the generic processes
- Numerical integration of the equations.

The report is in two volumes. The first describes the physics and numerics of the problem and the second describes the computer code. The first volume describes the salient features of the process to be simulated, the mathematical formulation of the problem, the choice of the numerical method of solution and the numerical formulation of the problem. The present status and recommended extensions are presented in the conclusion. The second contains user and programmer guides for the various phases of the simulation as well as compiled source code listings and load maps.

The computer code cannot at present describe the full purse seining process. The code can describe a single configuration at a time. As will be discussed in the next section, the seining process is a series of different configurations. An excellent example of the significant changes is the removal of the purse cable from the net. This cable, which closes and retrieves the bottom of the net, is split and removed from the net. This drastic structural modification removes the constraint that the net move with the cable and allows the net to be rolled in on the power block. This is a qualitative change in the problem and must be solved as one problem before the purse cable is removed and another after the cable is removed. Although each of these problems is of a form that can be solved by the code; the complex interfacing of the problems, the end of the first providing initial information for the second, has not been accomplished.

2. GEAR AND PROCEDURE

The net is the most significant part of the seining process. As shown in figures 1-4, the net is a complex object. Conceptually, there is the netting material itself, made of different types of net laced together. There is a corkline to float the net, a lead or chain line to sink it, a series of rings attached to the chainline for the purse cable to run through, and various lines permanently attached to the net and running through rings. The latter are zipper lines to divide the net into sections or bunch lines to gather sections of the net together.

The net as described is a well-defined entity. For the computer simulation, it is necessary to define the properties of the net once.

The manner in which the net is used, the procedure of purse seining, is also complex. The complexity here is not because the equipment is complex, but because the physical definition of the situation changes substantially at various times. A simple example of this is the recovery of the bow oertza and purse cable from the skiff; to the physical change there corresponds a mathematical change. The mathematical change is not a simple change in some physical parameter; it is the addition of an equation that relates the position of the bow oertza and purse cable to the seiner position. Thus, not only does the character of the equations change, but the number of equations to be solved changes. Although a packaged routine is used to solve the equations, one characteristic of this package relates closely to the changing number and structure of the equations and explains the requirement that the simulation of the whole seining process be considered as a series of seperate problems.

Numerical methods for solving the equations that describe the seining process must keep a record of how one part of the equipment acts on another part. The methods must know the effect

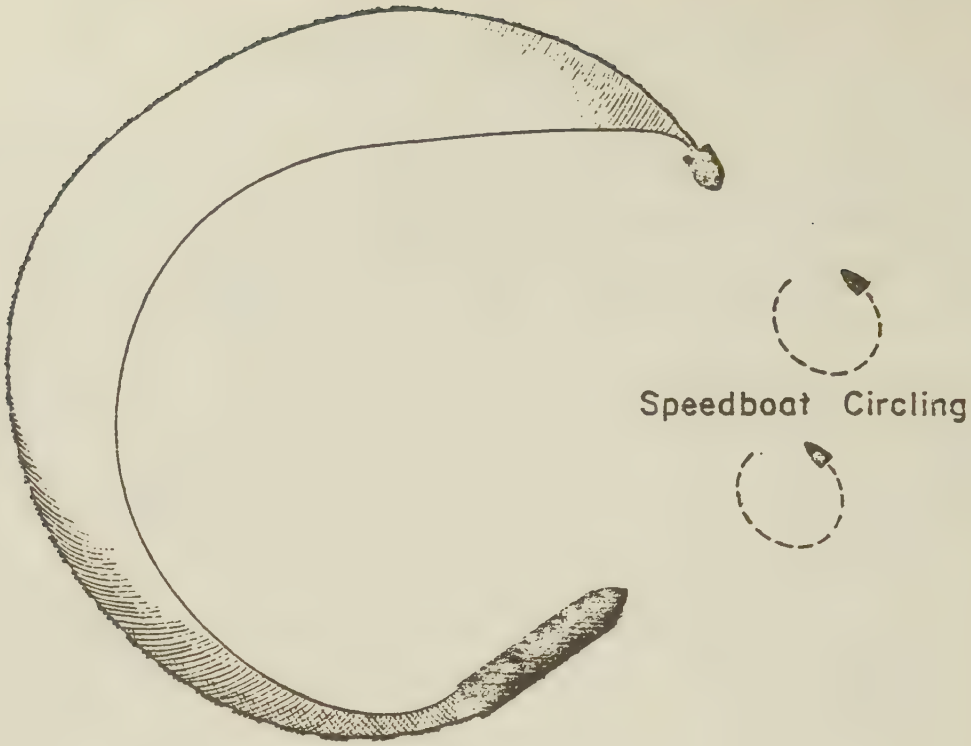


Figure 1a. Setting the Net.

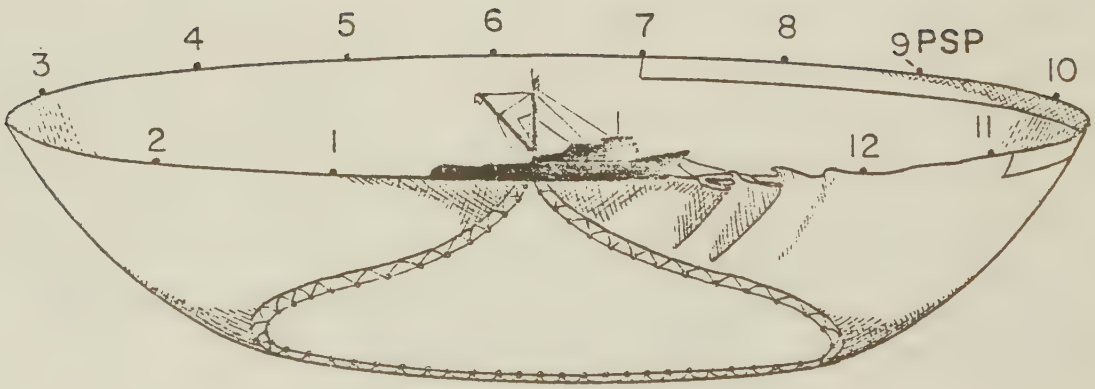


Figure 1b. Pursing (the net is about half pursed).

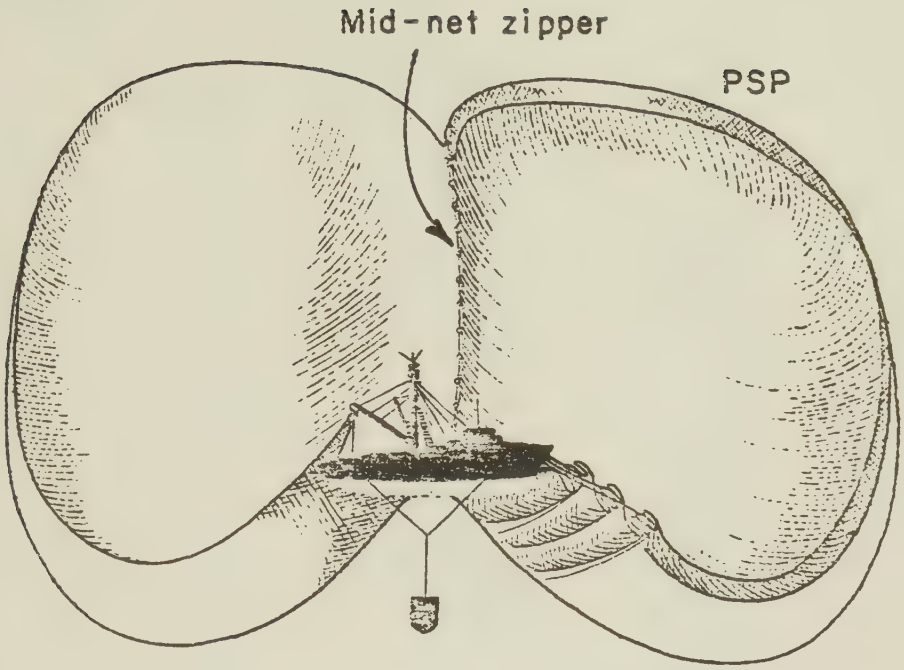


Figure 2a. Rings Up.

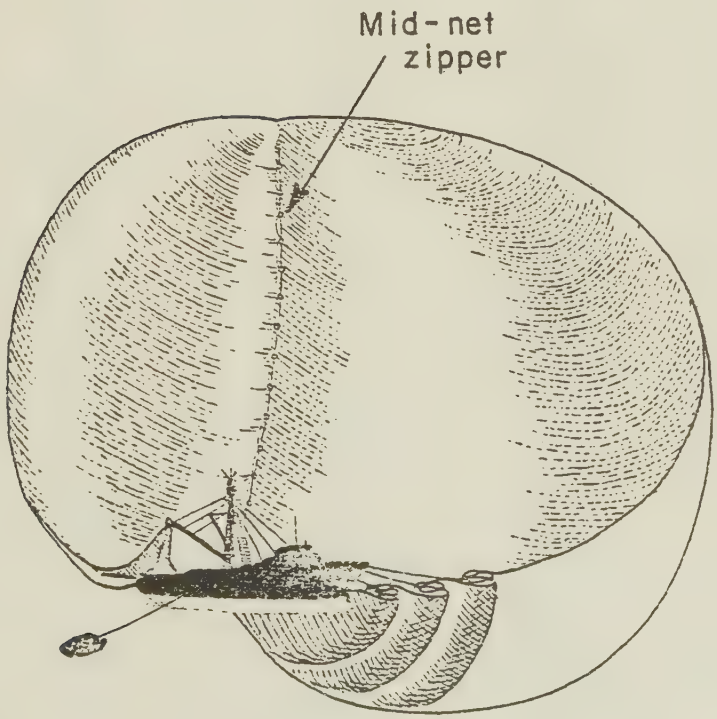


Figure 2b. Rolling the Net.

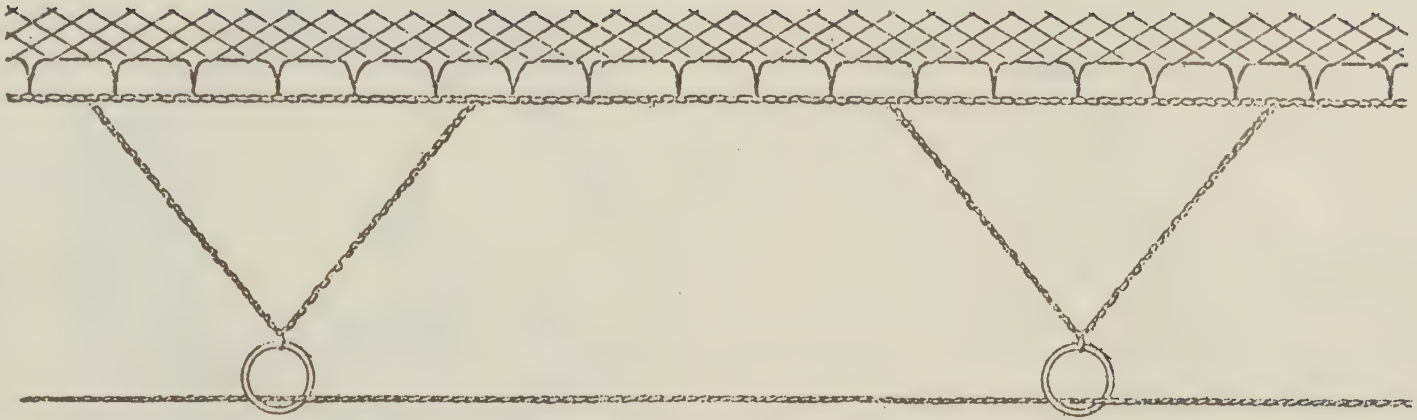


Figure 3. Purse Cable, Rings and Lead Line.

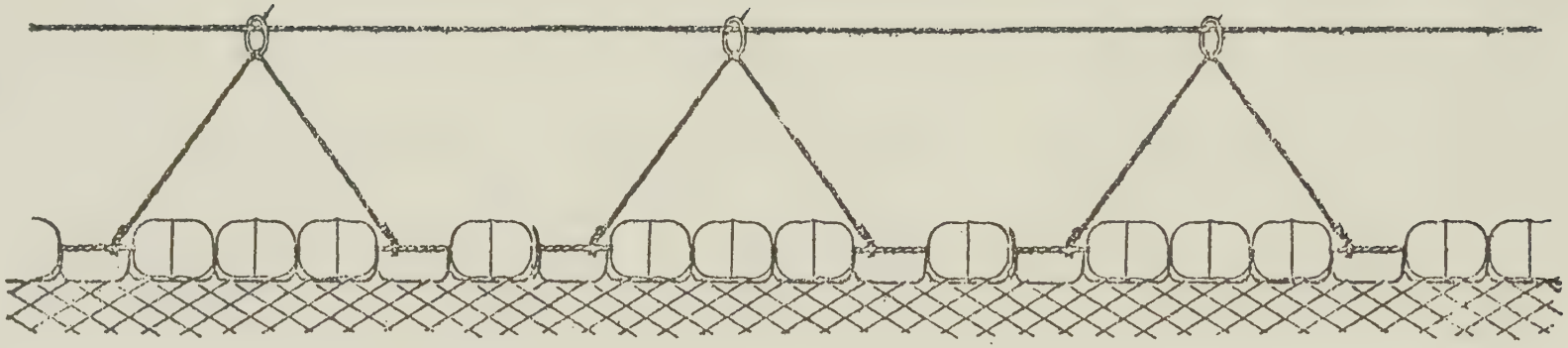


Figure 4. Cork Line, Bunch Lines.

moving one part has on other coupled parts. (This is the Jacobian matrix, c.f. Gear 1971). For instance, the purse cable acts on a ring, which acts on a part of the net, which acts on another part of the net, etc. The knowledge of how each element acts on adjacent elements is sufficient to tell how one part acts on all other parts of the gear. In general, any part of the net can be coupled directly to any other part of the net or, equivalently, any equation can be coupled to any other equation. If there are N equations, each one could in general couple to N-1 other equations giving a total of about N^2 interactions. Since each point is described by three velocities and three positions in three dimensional space, the total number of possible couplings is $(6k)^2$ where k is the number of knots. For two hundred knots there are more than one million possible couplings. This is too many to calculate or store.

The effective way to store and calculate the coupling characteristics is to keep a record of the coupled parts, i.e., which equations are coupled to which others. Since, on the average, each equation couples to about nine others, the storage requirements for real versus possible couplings is 9k versus $36k^2$, or about two thousand versus more than one million. This reduction is essential and is made by using GEARS, a code that uses special information on the coupling (section 4).

The method has the disadvantage that changing the coupling renders the previous coupling information useless. In the example, conceivably a problem could become totally coupled (the program must be able to handle all cases and thus also the general case) and the information required could increase by a factor of a thousand. Changing the coupling requires a restart of the calculation of the coupling and is equivalent to solving a new problem. For example, recovering the bow oertza and purse cable amounts to changing the problem, qualitatively adding some equations and dropping others. The old positions and velocities must be used as the initial values in the new problem. This interfacing of old and new problems has not been accomplished.

Tables 1-2 describe the seining process and the transitions between problem types that are necessary. The purpose is not to point out all that is taking place at a given time, but point out the features that define qualitative changes in the transitions to clarify the need for changes in problem description. These are generally changes of the physical configuration.

More specific descriptions of the various components of the gear are described in the next section.

Table 1. Phases of the Process that must be Considered as Seperate Problems.

Phase of the Seining Process	Delimiting Events
	Let 'er go
Setting the net	
	Bow oertza recovered from skiff
Pursing the net	
	Rings up
Roll in	
	Secure net for backdown
Backdown	
	Porpoise out of net/release net
Final roll in	
	Net fully stacked

Table 2. Special Features in Different Phases and Transition Characteristics

Phase/Transition	Special Features*
Transition to setting	Skiff holds bow oertza and purse cable in position
Setting	Net material forced to move with seiner until tension acts on it
	Purse cable/tow cable let out at specified rate
Transition to pursing	Bow oertza recovered and connected to seiner
	Bow end of purse cable connected to winch
Pursing	Purse cable rolled in at specified rate (both ends)
Transition to roll in	Rings up, on stripper. Cable attached to stern end of net over power block. Purse cable removed by opening split link
Roll in	Net at power block moves at fixed rate. Rings constrained to stripper.
Transition to backdown	Net at deck level gathered and constrained to move with side of seiner
Backdown	Seiner propulsion and thrusters activated
Transition to final roll in	Net released from side of seiner
Final roll in	Net at power block moves a specified speed until net fully stacked

*These features represent significant transitions in the number or types of equations that describe the process. This necessitates the restart of the solution. This has not yet been accomplished.

3. MATHEMATICAL FORMULATION OF THE PROBLEM

There are three generic parts to the net fishing process. In order of importance these are

- The net material
- Cables
- The boat.

The time evolution of these components is governed by equations of motion. These equations are described in the next three subsections. Although none of the equations is rigorous, sufficient background exists to indicate that the form of the equations is correct even if the parameters involved may require adjustment in specific cases.

3.1 EQUATIONS DESCRIBING NET MATERIAL

A mathematical description of the net is required for a numerical simulation. The net consists of material with various appendages. The equations governing the motion of the material are constructed here. The basis is the remark made by Kawakami (1964) that the mesh of a fishing net resembles a continuous membrane (figure 5). In the present case, the area of one square in the mesh is very small compared to the net area; the ratio of the areas is about one to one hundred million. This justifies the continuous membrane approximation. In fact, on a fine enough scale, real membrane once again resembles a mesh, but it is convenient to retain the concept of a continuous membrane. Of the four assumptions of Stengel and Leitzke (1975), two apply here. One is that the net offers no resistance to bending, and the other is that the fluid motion seen by the net is the free stream velocity; both of these assumptions will be good until the bars become nearly parallel. The resulting equations are partial differential equations. A survey of methods of solving these equations was made and the results are summarized in the section on

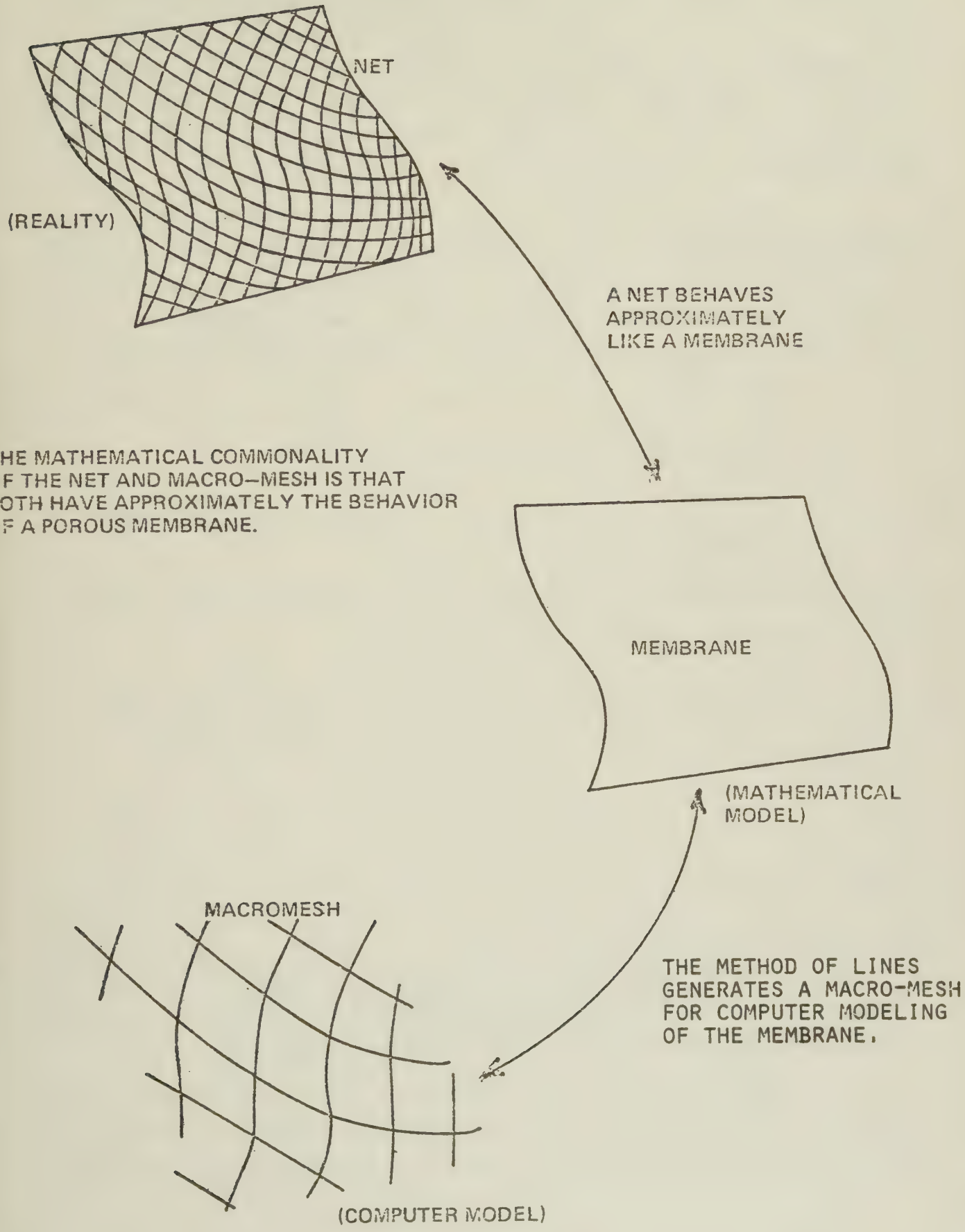

NET
(REALITY)
A NET BEHAVES
APPROXIMATELY
LIKE A MEMBRANE
THE MATHEMATICAL COMMONALITY
OF THE NET AND MACRO–MESH IS THAT
BOTH HAVE APPROXIMATELY THE BEHAVIOR
OF A POROUS MEMBRANE.
MEMBRANE
(MATHEMATICAL
MODEL)
MACROMESH
THE METHOD OF LINES
GENERATES A MACRO-MESH
FOR COMPUTER MODELING
OF THE MEMBRANE.
(COMPUTER MODEL)

Figure 5.

numerical methods. The selected method is the method of lines which is equivalent, in the present case, to having one unit in the numerical model represent a large number of bars. This transition to a macro-mesh is shown in the figure. The transition to the numerical model is described later. The real net and mathematical model are described here.

The net material may be considered as built up of a number of subunits as shown in figure 6. The equations governing the motion of such a subunit are written first and then a transition to partial differential equations is made. The choice of the numerical model and solution are described in later sections.

The relevant forces acting through the net material are

- Gravity and induced buoyancy
- Tension
- Viscosity

These forces combine to act on the mesh and accelerate it according to Newton's equations.

The acceleration due to gravity is a vector $\vec{g}$. Its effect is lessened by the buoyant force of displaced water so that the equivalent force is

$$(\rho_N V_N - \rho V)\,\vec{g} = m^*\vec{g}$$

where ρ_N, V_N are the density and volume of the net material, ρ, V are the density and displaced volume of water and m^* is the effective mass. For objects that float m^* is negative, but care must be taken to have the displaced volume, V, a function of vertical position if the object reaches the surface. In fact, floats are modeled as objects where V is such a function.

Tension forces are transmitted from adjacent parts of the net. Tension is due to the extension of the net material, and the knot positions must be known to describe it. The knots

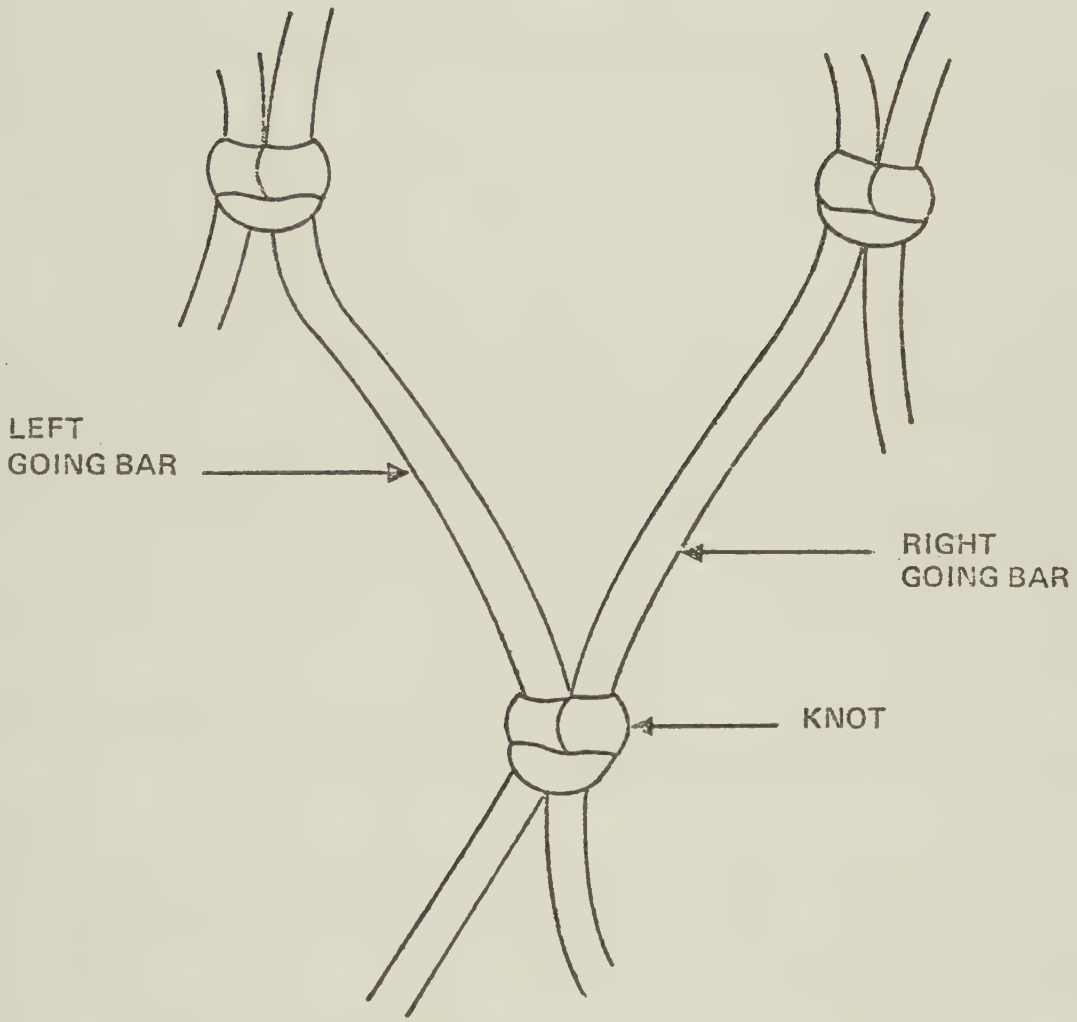


FIGURE 6. SUBUNIT OF A NET CONSISTS OF A KNOT AND TWO BARS. THE NET MATERIAL CAN BE CONSTRUCTED FROM SUCH SUBUNITS. THE REEF OR SQUARE KNOT SHOWN HERE IS ONLY USED IN JAPAN. MORE COMMON IS THE SINGLE WEAVER'S KNOT.

have position $\vec{x}(i,j)$ relative to the origin. Let i denote the position along right-aligned bars and j denote the position along left-aligned bars. Changing the index by unity denotes moving from one knot to the next. A useful notation is

$$\vec{D}(i) = \vec{x}(i+1,j) - \vec{x}(i,j)$$

$$\vec{D}(j) = \vec{x}(i,j+1) - \vec{x}(i,j)$$

with unit vectors

$$D = |\vec{D}|$$

$$\hat{D} = \frac{\vec{D}}{D}$$

It is convenient to denote the normal length of the bar as D_0. The tension is generally a function of the fractional extension of the material and acts along the vector extending from the knot under consideration to the next knot. The force due to the tension can be written

$$T(\frac{D}{D_0} - 1)\ \hat{D}$$

where T(x) is some function. Here a direct proportionality is used for $x > 0$ and no tension force for $x < 0$.

The most important force acting on the net is the reaction of the water. This force limits the sinking rate, the pursing rate, and all other motion of the net. To make a simple mathematical model of this force, several assumptions are required. These assumptions are discussed in this section and a useable form of the force is presented. The results are due to experimental work presented by Fridman (1969) and Koritzky (1977). Where the work is complete, the forces calculated here are accurate to 10%; where the work is incomplete, specifically for the lift of nets as a small angle to the flow, the forces are not known to 50%. However, the force in the last case is small. More complete experimental results should appear in the near future when some ongoing work described by Kruse (1977) is completed and interpreted.

The most important assumption is that the net can be considered as the sum of individual meshes (Fridman 1969, pg 62). A related assumption is that the net does not significantly modify the water motion. A case where these assumptions may not be valid is for a long net set parallel to the water flow. In this case a boundary layer may form and the resistance is not directly porportional to the area of the net (Fridman 1969, pg 66). This possibility is not considered here. It may be of importance late in the backdown process where the net has been pulled into a narrow channel. Even then, the motion of the net is somewhat circular so that the long boundary layer will not form.

Koritzky (1977) gives a procedure for calculating one component of the force on a net. He consideres the case where the angle between the flow and the bars are equal for all bars. The force in the direction of the flow is calculated by adding the forces on the individual bars and dividing by a correction factor. This is the drag force. There is also a force perpendicular to the relative velocity of the net and water called lift. Koritzky doesn't address this force and the lift force used here is just the sum of the forces on the bars.

In terms of dimensionless drag, parameters C_D and C_L, the force on a bar can be written

$$\vec{F}_D = \frac{\rho}{2} \vec{v}_R |\vec{v}_R| C_D A + \frac{\rho}{2} v_R^2 C_L A \frac{\hat{v}_R \times (\hat{v}_R \times \hat{u})}{|\hat{v}_R \times (\hat{v}_R \times \hat{u})|}$$

$$\vec{v}_R = \vec{v}_W - \vec{v}_B$$

where ρ is the water density, A the maximum projected area of a bar (length times diameter) $\vec{v}_W$ is the water velocity, $\vec{v}_B$ the bar velocity. Care must be taken with the last component of force since it represents lift. The sign of C_L will be the same as the sign of $\hat{u} \cdot \hat{v}$.

A reasonable fit, with an average error of 10% can be made to the data (Fridman, 1969, figures 31 and 32 for hemp, Koritzky, 1977 figures 2 and 6) with the approximations

$$C_L = C_1 \sin^2\gamma \cos\gamma$$

$$C_D = C_2 + C_3 \sin\gamma$$

where $\cos\gamma = \hat{u} \cdot \hat{v}$ and the dimensionless parameters

$$C_1 = 1.2$$

$$C_2 = .082$$

$$C_3 = 1.28$$

The coefficients C_L and C_D are plotted in figure 7. The constant, C_L, gives a 50% high zero angle drag, if one compares with the extrapolation in Koritzky, 1977, figure 2. The effect of knots can be approximated by increasing C_2.

The above shows no Reynolds number dependence, i.e., the coefficient C_L and C_D are independent of relative velocity. In fact, the variation of these coefficients is not important (20%) over a wide range of velocities, including the range of importance for nets (see Fridman 1969, figure 20). The drag coefficient for cylinders can be corrected for the net drag using Koritzky's (1977) method. The generalization from the symmetric flow is made here. The important parameter, k can be called the fractional self shadowing of the net viewed in the direction of the water. This is the area of the bars projected on the relative velocity divided by the area of the mesh projected on the relative velocity. (A factor of $\frac{1}{2}$ has been added in Koritzky's definition.) In the unsymmetric case this becomes

$$k = \frac{d}{2\ell} \frac{\left[|\hat{u}_1 \times \hat{v}_R| + |\hat{u}_2 \times \hat{v}_R|\right]}{|(\hat{u}_1 \times \hat{u}_2) \cdot \hat{v}_R|}$$

where the diameter of the bar is d and its length is ℓ. The subscripts refer to the two bars.

The assumption is that the same correction factor S(k) (Koritzky 1977, figure 1) holds for the generalized case as for the symmetric so that in the above

$$C_D \rightarrow C_D/S(k)$$

replacing the drag coefficient. The inverse of S(k) is plotted in figure 8. The effect is to increase the drag by up to 40% for a large amount of shadowing. (This effect is not presently included in the code).

The terms can be combined into an equation of motion for the mesh

$$\frac{d^2}{dt^2}(m_k+m_r+m_\ell)\,\vec{x}(i,j) = (m_k^*+m_r^*+m_\ell^*)\,\vec{g} + \vec{T} - \vec{F}_D$$

where

$$\vec{T} = T\left(\frac{D(i)}{D_0(i)} - 1\right)\hat{D}(i) - T\left(\frac{D(i-1)}{D_0(i-1)} - 1\right)\hat{D}(i-1) + T\left(\frac{D(j)}{D_0(j)} - 1\right)\hat{D}(j) - T\left(\frac{D(j-1)}{D_0(j-1)} - 1\right)\hat{D}(j-1)$$

and m_k, m_ℓ, m_r are the inertial masses of the knot, left-going and right-going bars. The effective inertial masses are increased in water because the acceleration of the material accelerates a mass of water. For cylinders the added mass is that of the displaced water. The net motion is not inertial so the effect of the added mass is small.

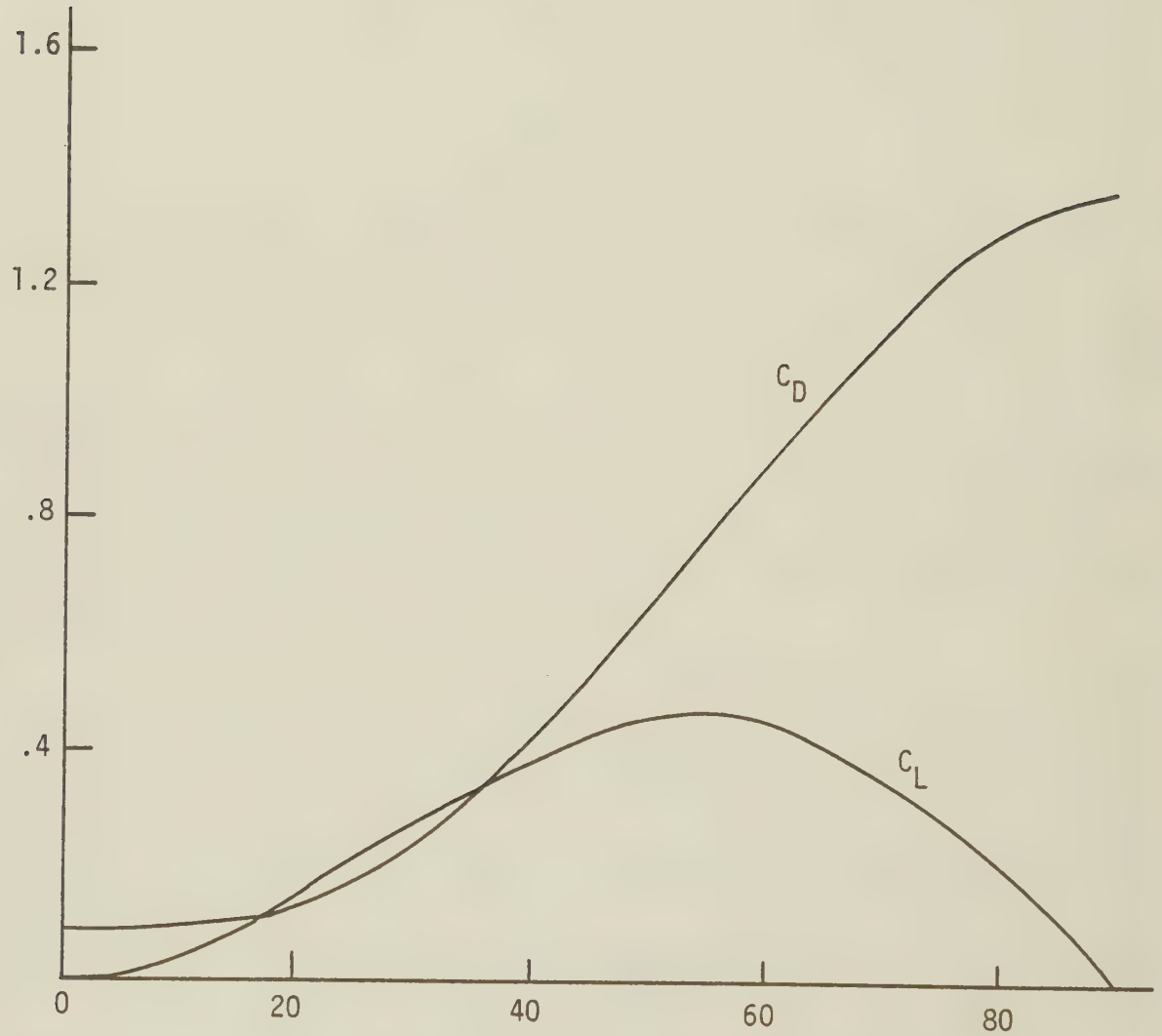


Figure 7. Drag and lift coefficients as a function of angle to flow for a hemp cylinder (from the analytical approximation).

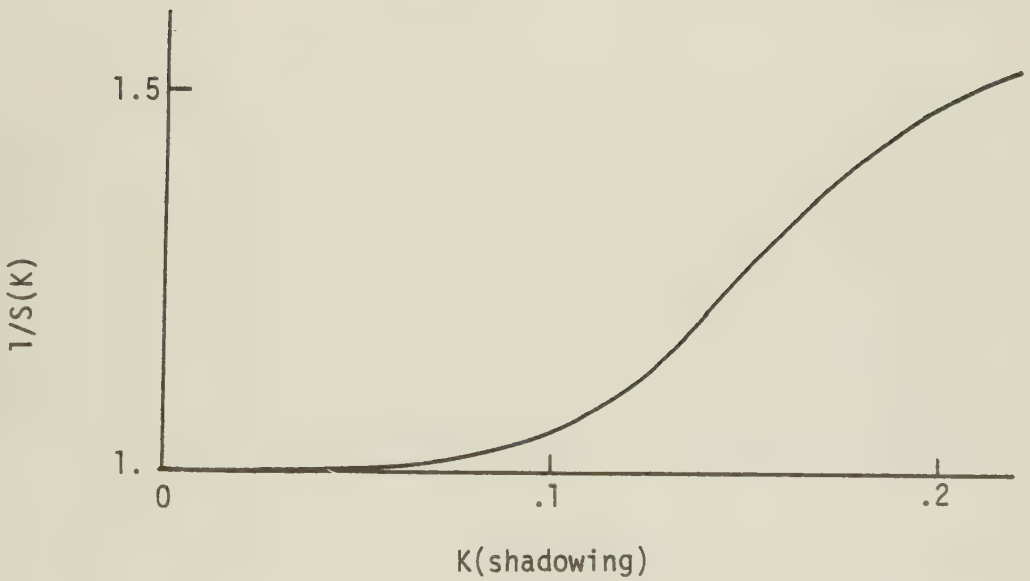


Figure 8. Multiplier (inverse of S(K)) for drag coefficient to correct for use as drag coefficient in nets.

The transition to membrane equations can be derived by introducing continuum coordinates in place of discrete indicies. Let

$$\zeta = i$$

$$\xi = j$$

Then the quantities can be approximated using

$$f(i+1,j) - f(i,j) \approx \frac{\partial f}{\partial \zeta}$$

so that the vectors are

$$\vec{D}(i) \approx \frac{\partial \vec{x}}{\partial \zeta}$$

and the tension becomes

$$\vec{T} \approx \frac{\partial}{\partial \zeta}\left\{T\left(\frac{D(\zeta)}{D_0(\zeta)} -1\right) \hat{D}(\zeta)\right\} + \frac{\partial}{\partial \xi}\left\{T\left(\frac{D(\xi)}{D_0(\xi)} -1\right) \hat{D}(\xi)\right\}$$

The replacement of the unit vectors carries over into the drag force so that the transition from discrete to continuous coordinates is complete. The boundary conditions are critical in defining any set of equations and this is true here also. The differential equations describe the material and the boundary conditions. Velocity, position, and tension determine the shape of the material and consequently must have the correct directions and magnitudes.

3.2 EQUATIONS DESCRIBING THE CABLES

The equations of motion describing cables attached to net by rings (c.f. figures 3-4) are not simple because the motion of the cable is forced to follow the net in the direction perpendicular to the length of the cable but free to move in the direction parallel to its length. The cable is tied to the net transversely but free longitudinally.

This type of motion is difficult to describe because the force between the cable and ring is unknown and must be a result of the calculation. This force, the constraining force, becomes what is necessary to keep the transverse motion of the ring and cable the same. Here this force is derived analytically for the ring and cable problem. It doesn't complicate the numerical solution.

One way to derive the equation of motion with constraints is to use Langrange multipliers. This is done here. The Lagrange density is used for the inertial part of the equation of motion. This is similar to the method used to get the differential equations to simulate the net motion (the cable is assumed to be totally flexible). The system is considered continuous and then approximated by discrete numerical elements (method of lines). Lagrange's techniques are not required to formulate the other forces. They have a simple form.

There are many implicit assumptions in deriving all of the equations of motion. It is not important to list these assumptions; it is important to look at the final equations and see if the significant forces are modeled correctly. From this point of view, the following gives a derivation of an important force: the force of constraint.

Some parameters are defined as:

$\vec{x}_c$ = cable position

α = distance to a point on the cable measured for the unstretched cable. (Thus, $\vec{x}_c = \vec{x}_c(\alpha)$.)

$\vec{x}_N$ = net or ring position

μ_c = cable mass per unit unstretched length

μ_N = ring (and leadline) mass per unit length

$\vec{u}$ = vector along cable = $\frac{\partial \vec{x}_c}{\partial \alpha}$

$\hat{u}$ = unit vector along cable = $\frac{\vec{u}}{|\vec{u}|}$

The inertial part of the Lagrange density is

$$L = \frac{1}{2} \mu_c \dot{\vec{x}}_c^2 + \frac{1}{2} \mu_N \dot{\vec{x}}_N^2$$

(Dots denote time derivatives.)

The equations of motion are obtained by minimizing

$$\int L \, d\alpha$$

with the positions $\vec{x}_c$ and $\vec{x}_N$ being the independent variables except as restricted by the rings.

The constraint is that the cable and rings undergo the same displacement perpendicular to the cable. This can be expressed by the equation

$$\hat{u} \times (d\vec{x}_c - d\vec{x}_N) = 0.$$

(Dividing this by dt shows that the relative transverse velocity is zero.) These are three equations in three unknowns. One of these is redundant because the restriction is only in two dimensions. This is clear because adding a multiple of $\hat{u}$ to any solution for $d\vec{x}_c$ gives another solution.

Lagrange's way of incorporating these restrictions into the equations of motion is to multiply each equation by an unknown multiplier and add to the variation. The minimization principle is then

$$\int \left[\delta L - \vec{\eta} \cdot \hat{u} \times (d\vec{x}_c - d\vec{x}_N)\right] \, d\alpha = 0$$

Now, however, the positions are considered independent. The unknown multipliers, $\vec{\eta}$, are then chosen so that the displacements satisfy the constraints.

The equations of motion are now

$$\frac{d}{dt}\,\mu_c\,\dot{\vec{x}}_c + \vec{\eta} \times \hat{u} = \vec{F}_c$$

$$\frac{d}{dt}\,\mu_N\,\dot{\vec{x}}_N - \vec{\eta} \times \hat{u} = \vec{F}_N$$

with the constraint

$$(\dot{\vec{x}}_c - \dot{\vec{x}}_N) \times \hat{u} = 0$$

These are nine equations in nine unknowns. The $\vec{F}_c$ and $\vec{F}_N$ are forces per unit length other than those of constraint. This shows clearly that $\vec{\eta} \times \hat{u}$ is a force per unit length acting on the rings.

Some manipulation of these equations clarifies their meaning and puts them into a useful form. Adding the first two gives the total momentum equation

$$\frac{d}{dt}\,(\mu_e\,\dot{\vec{x}}_c + \mu_N\,\dot{\vec{x}}_N) = \vec{F}_c + \vec{F}_N$$

so that the forces of constraint disappear. They are equal and opposite.

The relative acceleration is interesting because it is constrained. If the ring mass is constant, but the cable mass changes (as when a larger diameter section of purse cable enters a ring), the relative acceleration is

$$\frac{d}{dt}(\dot{\vec{x}}_c - \dot{\vec{x}}_N) + \vec{\eta} \times \hat{u}\left(\frac{1}{\mu_c} + \frac{1}{\mu_N}\right) = \frac{\vec{F}_c}{\mu_c} - \frac{\vec{F}_N}{\mu_N} - \frac{\dot{\mu}_c}{\mu_c}\,\dot{\vec{x}}_c$$

This can be used to determine the Langrange multiplers, $\vec{\eta}$. First differentiate the constraint equation to give

$$\dot{\hat{u}} \times (\dot{\vec{x}}_c - \dot{\vec{x}}_N) + \hat{u} \times \frac{d}{dt} (\dot{\vec{x}}_c - \dot{\vec{x}}_N) = 0$$

Next, note that any component of $\vec{\eta}$ in the direction of $\hat{u}$ has no influence on the problem since

$$(\vec{\eta} + \hat{u}) \times u = \vec{\eta} \times \vec{u}$$

so choose

$$\vec{\eta} \cdot \vec{u} = 0.$$

This is the reduction, by one, of the number of multipliers. This fits because there were three equations for a two dimensional constraint, and now there are two.

Taking the cross product of the equation of relative motion with $\hat{u}$ and using the constraint to eliminate the relative acceleration gives

$$\vec{\eta} = \frac{\mu_c \mu_N}{\mu_c + \mu_N} \left[\dot{\hat{u}} \times (\dot{\vec{x}}_c - \dot{\vec{x}}_N) + \hat{u} \times \left(\frac{\vec{F}_c}{\mu_c} - \frac{\dot{\mu}_c}{\mu_c} \dot{\vec{x}}_c - \frac{\vec{F}_N}{\mu_N} \right) \right]$$

This can be further simplified by noting that the relative velocity is along $\hat{u}$ so

$$\dot{\vec{x}}_c - \dot{\vec{x}}_N = v_c \hat{u}$$

where v_c is the cable speed relative to the net. Note further that, since $\hat{u}$ is a unit vector any change in $\hat{u}$ must be a rotation so that

$$\dot{\hat{u}} \cdot \hat{u} = 0$$

Using this to get the force of constraint acting on the rings gives

$$\vec{\eta} \times \vec{u} = \frac{\mu_c \mu_N}{\mu_c + \mu_N} \left(\frac{\vec{F}}{\mu} - \hat{u} \hat{u} \cdot \frac{\vec{F}}{\mu} - \dot{\hat{u}} v_c \right)$$

where

$$\frac{\vec{F}}{\mu} = \frac{\vec{F}_c}{\mu_c} - \frac{\dot{\mu}_c x_c}{\mu_c} - \frac{\vec{F}_N}{\mu_N}$$

Substituting this back into the equation for the relative acceleration, and using v_c in place of $\dot{\vec{x}}_c - \dot{\vec{x}}_N$ gives

$$\frac{d\, v_c}{dt} = \hat{u} \cdot \left[\frac{\vec{F}_c}{\mu_c} - \frac{\dot{\mu}_c\, \dot{\vec{x}}_c}{\mu_c} - \frac{\vec{F}_N}{\mu_N} \right]$$

This says, except for the mass derivative, that the relative acceleration is just the difference between the cable and ring acceleration along the direction of the cable.

The equation of motion for the ring position is

$$\frac{d\, \dot{\vec{x}}_N}{dt} = \frac{1}{(\mu_c + \mu_N)} \left\{ \vec{F}_N + \frac{\mu_c}{\mu_N} \hat{u}\, \hat{u} \cdot \vec{F}_N + \vec{F}_c - \hat{u}(\hat{u} \cdot \vec{F}_c) - \mu_c\, \dot{\hat{u}}\, v_c \right\}$$

The force terms in this equation can be interpreted by denoting the component parallel to $\hat{u}$ with the subscript $\parallel$ and the component perpendicular to $\hat{u}$ with the subscript $\perp$. The equation becomes

$$\frac{d\, \dot{\vec{x}}_N}{dt} = \frac{\vec{F}_{N\perp} + \vec{F}_{c\perp}}{(\mu_c + \mu_N)} + \frac{F_{N\parallel}}{\mu_N} - \frac{\mu_c\, \dot{\hat{u}}\, v_c}{(\mu_c + \mu_N)}$$

Thus the perpendicular forces see the combined masses while the parallel component of the cable force doesn't appear and the parallel component of the force on the rings sees only the ring mass.

To take into account friction, the constraining force that is normal to $\hat{u}$ is $\vec{\eta} \times \hat{u}$ and

$$F = |\vec{\eta} \times \hat{u}|$$

is the quantity to use in the friction equation

$$\vec{f}_N = \text{sign}(v_c)\, \sigma\, F\, \hat{u}$$

for a coefficient of friction σ.

The above almost completes the equations of motion; the equation for $\dot{\vec{x}}_c$ is

$$\frac{d\dot{\vec{x}}_c}{dt} = \frac{1}{\mu_c + \mu_N}\left[\vec{F}_c + \frac{\mu_N}{\mu_c}\hat{u}\,\hat{u}\cdot\vec{F}_c + \vec{F}_N - \hat{u}\,\hat{u}\cdot\vec{F}_N\right.$$

$$\left. - \frac{(\mu_c + \mu_N)}{\mu_c} v_c\,\dot{\mu}_c\,\hat{u} + \mu_N\,\dot{\hat{u}}\,v_c\right]$$

However, as will be seen later in the application of this equation of motion, the best choice is to solve for v_c and $\vec{x}_N$, rather than $\vec{x}_c$ and $\vec{x}_N$.

Furthermore, to be able to calculate the tension in the cable, the stretched length, β, of the cable to a point $\vec{x}_N$ as well as the unstretched length α are required. The equation of motion for these quantities are

$$\dot{\beta} = -v_c$$

$$\dot{\alpha} = \dot{\beta}/\frac{\partial\beta}{\partial\alpha}.$$

The tension is a function of the fractional stretch $\partial\beta/\partial\alpha$. (In the computer code $\partial\beta/\partial\alpha$ is assumed to be unity and tension is computed from $\partial|x|/\partial\alpha$.)

These equations will be summarized in the section on applications.

Finally, there is one point that, although not relevant to constrained motion, is important for the cables. The point is that the cables are constrained only over part of their length. For instance, the section of the purse cable between the leadline and the seiner is free so there is no natural reference point; that is, there is no $\vec{x}_N$. An approach to providing a reference point is to hang some 'artificial' rings on this part of the cable. These have a repulsive force between them to keep them reasonably spaced and a damping term to keep them from oscillating. The acceleration on one of these imaginary rings parallel to the cable is

$$a_r = k\,\frac{\partial^2\alpha}{\partial_{\text{index}}} - \sqrt{8k}\,\frac{\partial\dot{\alpha}}{\partial_{\text{index}}}$$

where index is the index of the ring. At the seiner the boundary condition is $\dot{\beta}$ and the first ring position is the other boundary.

3.3 EQUATIONS DESCRIBING THE SEINER

There are two phases in the seiner motion. The first phase is the high speed phase while the net is being set and the second is the low speed phase that follows. In the first phase the seiner is free to maneuver so long as it closes the path at the skiff. In the second phase the seiner is tied to the net and no longer free to maneuver. During the first phase the path of the seiner can be prescribed because the net does not significantly effect the motion of the seiner. The path, prescribed by a forward speed and specified angular velocity, must only conform to performance characteristics for speed and maneuverability of the boat.

During the second phase the net and seiner motion are intimately coupled. Furthermore, most of the significant parameters of the net such as closing time and shape are determined during this phase. The motion of the seiner under the influence of various forces is calculated for this phase using certain simplifications that apply for low speed motion (Comstock, 1967).

The equations of motion of the ship are most simple in coordinates with origin at the center of gravity of the ship. The ship moves in the plane of the surface and has a velocity u along its axis and v perpendicular to its axis. The angle that the axis makes with the x axis is θ, increasing θ moves a right hand screw up the z axis which is upward (opposite from usual naval architect's coordinates).

The seiner motion is specified until reaching the skiff. After this, the motion is determined by integrating the equations of motion. The forces on the boat accelerate it in the u and v directions and create a torque that changes θ. This means that every force must have a magnitude, direction and point of application that contribute to two forces, F_u along the axis, F_v perpendicular to the axis, and a torque N_z about the center of gravity.

In the following, the coordinates are defined, the forces formulated and the equations of motion presented.

The coordinates and velocities are defined in Table 3. All angles are measured from the x axis. Some of the information is most conveniently specified in a coordinate system relative to the center of gravity of the ship. Thus the tension (Table 4) on the purse cables acts at a point that is always at a fixed position relative to the center of gravity of the seiner. The propulsion also acts at a point that is at a fixed position relative to the center of gravity of the seiner and, additionally the direction of the force is specified (Table 5). Table 6 also shows the necessary information for the calculation of the contribution of the tension and propulsion to the force. Since the seiner velocity is measured relative to the seiner, the forces must also be in those coordinates. The forces F_u, F_v and torque N_θ about the verticle line through the center of gravity are accumulated by summing the contributions δF_u, δF_v, δN_θ from the various forces. An example of the procedure for calculating these contributions from a tension is shown in the table.

Table 3. Coordinates for Seiner

Ship		Wind	Current
Absolute	Relative		
x_s	u = speed along ship axis	v_w = speed	v_c = speed
y_s	v = speed along ship axis	θ_w = angle to x axis	θ_c = angle to x axis
θ_s = angle of ship axis to x axis			

Table 4. Tension Forces

Reference point = $\vec{x}'_R$ or $\vec{x}_R$(relative to ship)

Unit vector along tension away from ship = $\hat{U}_R$

Tension = T

Reference relative to ship

$$\vec{x}'_R = \vec{x}_S + \vec{x}_R$$

$\vec{x}_S$ = absolute position of ship

Table 5. Propulsion Forces

Reference point	$\vec{x}'_R$	$\vec{x}_R$ relative to ship
Unit vector	$\hat{u}'$	$\hat{u}$ relative to ship (time history specified)
Force	F	(time history specified)

Table 6. Components of Force and Torque

Tension and Propulsion

$$\theta_R = \text{atan2}(\hat{u}_{Ry}, \hat{u}_{Rx})$$

$$\delta F_v = T\cos(\theta_s - \theta_R)$$

$$\delta F_u = T\sin(\theta_s - \theta_R)$$

$$\delta_{N\theta} = (\vec{x}_R \times T\hat{u}_R)_{z \text{ component}}$$

The wind and current forces on the seiner are calculated assuming the seiner speed is small compared to the wind and current speeds. The contribution from the wind is approximately

$$\delta F_u = \frac{1}{2} \rho_a C_{xw} V_w^2 A_{px} \cos(\theta_w - \theta_s)$$

$$\delta F_v = \frac{1}{2} \rho_a C_{yw} V_w^2 A_{py} \sin(\theta_w - \theta_s)$$

$$\delta N_\theta = x_{CA} \delta F_v$$

where ρ_a = air density

$A_{\rho x}$ = projected area, front direction

$A_{\rho y}$ = projected area, side direction

x_{CA} = distance of center of area from center of gravity.

The dimensionless constants are approximately

$$C_{xw} = .6$$

$$C_{yw} = .87$$

as for a tanker and reflect the more streamlined shape in the forward direction.

The force due to currents at low speed is given by Remery and Van Oortmerssen (1973). They are similar in form to the above equations

$$\delta F_u = \frac{1}{2} \rho V^2 C_{xc}(\theta_w - \theta_s) A_{TS}$$

$$\delta F_v = \frac{1}{2} \rho V^2 C_{yc}(\theta_w - \theta_s) A_{LS}$$

$$\delta N_\theta = \frac{1}{2} \rho V^2 C_{NC}(\theta_w - \theta_s) A_{LS} \cdot L$$

A_{TS} = submerged transverse hull area

A_{LS} = submerged lateral area = LXT

T = draft of seiner

L = length of seiner

ρ = water density

V = water velocity

The coefficients are more complicated now, given by

$$C_{xc}^{(\alpha)} = \frac{.075}{(\log_{10}(R_n)-2)^2} \frac{S}{A_{TS}} \cos(\alpha) \, |\cos(\alpha)|$$

$$C_{NC}^{(\alpha)} = \sum_{i=1}^{5} a_i \sin i\alpha$$

$$C_{yc}^{(\alpha)} = \sum_{i=1}^{5} b_i \sin i\alpha$$

$$R_N = \frac{V \cos\alpha}{\nu L}$$

The Reynolds number, R_N, is a function of water viscosity ν. Note that in this case, S, the wetted surface area is important, the transverse area cancelling. The coefficients a_i, and b_i are given in the Table 7.

Table 7. Coefficients a and b for Water Drag on Seiner

i	a_i	b_i
1	-.0252	.908
2	-.0904	.000
3	.0032	-.116
4	.0109	.000
5	.0011	-.033

The combined force can be used in the equation of motion of the seiner. It is assumed that the seiner moves in the horizontal plane of the ocean surface. The equations of motion with the vertical axis passing through the center of gravity are given in Table 8. This form is used directly in the differential equation integration.

Table 8. Equations of Motion of the Seiner

Variable	Time Derivative
x_s	$u \cos\theta_s - v \sin\theta_s$
y_s	$u \sin\theta_s - v \cos\theta_s$
U	$(F_u + M \dot{\theta}_s V)/M$
V	$(F_v - M \dot{\theta}_s U)/M$
$\dot{\theta}_s$	$\theta_s N/I_z$
θ_s	$\dot{\theta}_s$

M is the seiner mass and I_z is the moment of inertia around the verticle axis through the center of mass.

One inadequacy of the formulation is that no damping takes place for a ship in pure rotation. Naval architects are not interested in this case so that this doesn't occur in the classical literature. It is, nevertheless, disconcerting to have the seiner cease moving in either forward or sideways directions but continue to rotate about its axis indefinitely. An ad hoc method is used here to estimate a drag. The force is due mostly to flow transverse to the seiner. The assumption is made that the torque can be estimated from the following:

$$N_z \approx \frac{-1}{2} \rho V_{rms} \int_{-L/2}^{L/2} V \ell d\ell \, T$$

where T is the draft; V_{rms} is the root mean square of the perpendicular water velocity; ℓ is the moment arm. These quantities can be evaluated to give

$$N_z \approx \frac{-1}{2} \rho \sqrt{\frac{(L\dot{\theta})^2}{12} + V^2} \; \frac{L^3 \; \dot{\theta} \; T}{12}$$

For the case of large perpendicular velocity, V, this agrees well with standard naval formulae but does not reduce to zero for V = 0.

4. MODELING PROCEDURE AND NUMERICAL METHODS

Simulation is based on a model. A model is a numerical representation of the mathematics that describes the behavior of an idealized physical system. A computer can work with a numerical model. The way the computer works with the model is the numerical method. Important points relating to modeling and numerical methods are

- Modeling procedure
- Accuracy of the model
- Selection of numerical methods
- Potential problems.

In this section, these points are discussed in light of results available in the literature. This discussion summarizes the motivation and justification for the model and numerical method chosen for simulating net fishing.

4.1 NUMERICAL MODEL

The net and cables are described by partial differential equations (the seiner is considered as a rigid body and is described by ordinary differential equations) with continuity in both space and time. The chosen method for translating the partial differential equations into equations solvable by a computer is the method of lines. This method is simple to formulate because it considers that, along a line between two points, the parameters such as drag and velocity are constant. There are other methods, finite element methods, that consider the variables as changing between two points with some functional dependence. These are much more sophisticated and difficult to implement. (In fact, the method of lines was considered until recently as one of the finite element methods; it became an outcast because of its simplicity.) Hopkins and Wait (1978) compare several of these methods on a number of problems. The problems are parabolic partial differential

equations. This is a class of problem similar to the net problem in that both have considerable dissipation and don't propagate waves. In a net this is equivalent to the fact that the most significant force on the net when it is perturbed is the drag of the water.

It can be seen from the results of Hopkins and Wait that the method of lines was more economical in computer time than the other methods for comparable accuracies.

In addition to being simple to formulate and reasonably accurate, the model results in a set of ordinary differential equations for which computer solution packages are readily available, well understood, and extensively documented. The method of lines is ideally suited to the present problem.

4.2 ACCURACY OF THE MODEL

One way of defining accuracy in the seining problem where the configuration of the net is important, is the fractional error in relative position of various parts of the net. This is a fairly direct measure of the shape of the net. For a slack net, the knots can be up to a bar length in error. However, in the more important case where the net is being pulled through the water, the error will be significantly less.

A net under tension looks like a hanging cable in cross section. A view along a sequential set of left or right going bars shows a cable loaded at the knots by the right or left going bars. Wang (1975) has analysed the error that occurs when the method of lines is used to describe a loaded cable in the static (or steady state) case. His model 2 most closely resembles the present model. For the problem he considered, the average error in the approximate position (average of distance from calculated to exact position) and error in tension are shown in Table 9 for different numbers of segments.

Table 9. Error When Using the Method of Lines

Number of Segments	Position Error %	Tension Error %
2	5.11	9.66
5	1.92	2.72
10	.62	.69

The accuracy that can be expected in the present case is substantially less because of the two dimensional nature of the problem. A priori, one would expect an error for the cable problem given by

$$\text{position error } \% \approx 100/\text{number of segments.}$$

Obviously, this is a particularly pessimistic estimate when compared to Wang's results and more accurate results are likely.

The above has addressed the accuracy of the model used to translate the partial differential equation into a numerical model. The accuracy of the total simulation is dependent on many factors and the accuracy of the model is only one of these.

4.3 SELECTION OF NUMERICAL METHODS

The modeling procedure gives a set of time dependent coupled ordinary differential equations. A numerical method must be chosen to integrate these equations, and the method depends on the characteristics of the equations. The salient characteristics of the set of differential equations describing the purse seining process is that they are stiff. This characteristic can be described by the fact that certain things, such as a change in the extension of a purse cable or the net twine, would relax in a very short time (milliseconds as exemplified by the snapping of the purse cable when it is rolled onto the winch). However, the motion of the cable and net are rather smooth and continuous and

take place over a long time. (If the purpose were to model the snapping, significant changes would occur on small time scales and the solution would not be stiff.) This disparity of time scales and the fact that no significant changes occur on the small time scales is a qualitative definition of stiffness.

Another salient feature of the problem is that the quantities that are most of the unknowns, the knot positions and velocities, don't interact with many other quantities directly. For instance, most knots directly influence the four surrounding knots only and only influence distant knots indirectly. This means that the Jacobian matrix (the record of which unknowns influence which others and how much) has a large number of zeros. Such matricies are called sparse matricies. Efficient methods of handling data eliminate the need to store the zeros and save a large amount of computer storage.

Fortunately, stiff problems with sparse Jacobians occur frequently in modeling physical problems, and a great deal of high quality work has been done on solving them. A review by Shampine and Gear (1979) provides a useful look at solving such equations from the users point of view and provided the basis for the approach used here. Dr. Alan Hindmarsh of Lawrence Livermore Laboratory has kindly supplied a copy of GEARS, a computer code that uses the methods described by Shampine and Gear (Gear, 1969, is a major contributor to the efficient solution of stiff equations) and combines them with the Yale sparse matrix package. This code has been used to integrate the equations describing the seining process.

4.4 POTENTIAL PROBLEMS

The accurate numerical solution to ordinary differential equations becomes difficult when there are discontinuities in the derivatives. Physically this occurs when there is a snap-in, i.e., when a cable or net twine goes from slack to being taut. The problem also arises in the earthquake response of structures when

losely spaced walls collide. Carver (1979) investigated the atter problem and found that the method used in GEARS does not erform too poorly and he gives a method of improving the speed f computing over a discontinuity by a factor of two. His method as programmed and performed as he described (but was not inte- rated into the overall simulation code).

The procedure can be summarized simply. The predictor art of GEARS is used to predict when a discontinuity will occur. f the prediction is within the next time step, the prescription s to integrate just up to the discontinuity, use a small time tep to cross over it and restart with a large time step. The ssence of the method is to avoid the search for the discontinuity y hit or miss that usually occurs because of accuracy criteria nd further to avoid the slow increase in time step from the small alue required to find the discontinuity.

However, in the full scale simulation snap-in may not be problem. This is probable because the effect would be both physi- lly and computationally traumatic. If it were to occur to such a egree as to be significant on a full scale simulation, it would ndicate that the system is loaded sufficiently to fail. Since this oes not happen regularly in reality, it would not occur in an ccurate simulation.

Nevertheless, the purpose of the simulation here is not tructural stress, but geometric response. It is important, though, o have a method for overcoming the problem if it arises.

.5 GENERAL COMMENTS

The information in the literature substantiates the fol- owing conclusions about the method chosen here to describe the eining process:

- The method is simple
- The method is accurate
- The numerical method is reasonably economical in computer storage and time.

The numerical method takes advantage of sparse matrix techniques to store efficiently the information on how the various parts of the system are coupled. However, this method, which reduces the required calculation by several orders of magnitude, requires that the coupling information be recalculated everytime a substantial change in structure occurs. This is summarized in the Tables 1 & 2.

However, there are other anomalies in the process that require special treatment, i.e., phenomena that cannot be described by simple ordinary differential equations. One such phenomenon is the setting of the net and the treatment is briefly mentioned here to point out the difficulty that arises in a class of problems and a typical solution.

A section of net moves with the seiner until it is pulled off the seiner. Thus, before the section of net is pulled off, it is constrained to move with the seiner; its velocity is that of the seiner. Afterward, the section of net moves according to a set of equations of motion. This is a substantial change in the equation of motion and the coupling due to this latter configuration must be stored in the sparse matrix structure to avoid the necessity of restarting the problem. This amounts to storing the possible coupling even though before the net is pulled off the seiner the amount of coupling is zero. However, the main point here is that the system must tell when to move the net with the seiner and when to move the net according to a Newtonian equation of motion. The answer is to use a set of hysteresis functions for each knot. This function was chosen to be the time integral of the tension acting on a knot. Until this quantity reaches a certain value, the knot moves with the seiner. Afterwards the knot moves as described by the equations of motion.

Such functions allow the solution to continue smoothly over variations that do not change the basic coupling in the system, but do change the form of the equations of motion.

5. NUMERICAL MODEL

It was determined in the last section that the method of lines is the best method for constructing a model. The method is to assume that variables are constant between two points. There is some difference in the way in which the partial differential equations for nets and cables are translated into ordinary differential equations so the terms will be discussed separately.

5.1 NUMERICAL MODEL OF THE NET

The unit element in the macro mesh represents a rhombic section of mesh in the numerical model with $d\zeta$ bars in the right aligned direction by $d\xi$ in the left-aligned direction. The $d\zeta$ and $d\xi$ values are the scaling factors. This area has a mass of

$$M = d\zeta \; d\xi (m_k + m_r + m_\ell)$$

and a drag

$$\vec{F}_D \Rightarrow d\zeta \; d\xi \; \vec{F}_D$$

The tension is assumed constant over the boundary so that the tension represents the force due to $d\zeta$ bars in the left-aligned direction and $d\xi$ bars in the right-aligned direction. The unit vectors are between two centers of the macro mesh and the vectors are now between a pair of knots a large distance apart.

$$\vec{D}(i) = \vec{x}(i + d\zeta) - \vec{x}(i)$$

Thus, the derivative tension becomes

$$\begin{aligned}
\vec{T} \Rightarrow\; & d\xi \; T\left(\frac{D(i)}{D_0(i)} - 1\right) \hat{D}(i) \\
& - d\xi \; T\left(\frac{D(i-d\zeta)}{D_0(i-d\zeta)} - 1\right) \hat{D}(i-d\zeta) \\
& + d\zeta \; T\left(\frac{D(j)}{D_0(j)} - 1\right) \hat{D}(j) \\
& - d\zeta \; T\left(\frac{D(j-d\xi)}{D_0(j-d\xi)} - 1\right) \hat{D}(j-d\xi)
\end{aligned}$$

The simple approximation to the derivative used here makes the equations look like the net equations for a large, or macro mesh. This would indicate that a more common sense, rather than mathematical, basis exists for the construction of a macro mesh model. However, the fact that a partial differential equation describes the net and that there is literature justifying the choice of the method are reason enough to use the intermediary of the partial differential equations.

The important points are that the mass and drag scale as the scaled area, $d\zeta\, d\xi$, and the tension scales as the transverse scaled distance $d\xi$ or $d\zeta$.

5.2 NUMERICAL MODEL OF CABLES

The cable equations are simple to translate into numerical ordinary differential equations because they are already formulated in quantities per unit length. This means that the quantities in the equations do not depend on the length factors in the method of lines. The unit vectors are chosen between the ring positions, but the cable velocity is assumed to be along the vector between the two adjacent ring positions except at the end points. These equations are summarized in tables.

For the rings, the quantity $\partial\beta/\partial\alpha$ was set to unity because the cables do not stretch considerably. It was also found that the artificial time constant, k, could be set to $1/\text{sec}^2$. The damping can be calculated more simply in this case to be

$$\sqrt{8}(\dot{\alpha} + v_c)$$

where v_c is the average of the cable velocities at the right and left hand rings.

5.3 NUMERICAL FORMULATION OF THE EQUATION OF MOTION OF A CONSTRAINED CABLE

There are three kinds of cables that are constrained to move with the net. These are described in table 10 and figures 1, 2, 9 and 10.

The two types of equations that describe the motion of a cable

1) freely hanging between the net and a specified point, and

2) constrained to move transversely with the net by rings,

will be combined to describe the motion of the three kinds of cables.

The numerical method to solve these equations will be outlined here. The derivation of the equations is given in the previous section. The method is chosen to handle an important aspect of the problem: the change in length of the significant part of the cables during the simulation. An example is the purse cable. At the beginning of pursing of the net, the purse cable interacts with the net over a length of cable greater than one thousand meters. At the time when the rings are up, the length of the cable that intereacts with the net is about 10 meters.

It is necessary to make the continuous equations of motion into a descrete set of equations for numerical solution. The fixed points for the discrete set can be chosen as points moving with the cable or points fixed at points where the cable and the net interact. If the points are chosen to move with the cable, most of them would move out of the region where the cable interacts with the net. This would not give a very accurate description of the net. The other choice is the prefered one because the number of points where the cable interacts with the net won't change and the accuracy of the description won't decrease.

Table 10. Types of Constrained Cables

Purse Cable	Bunch Lines	Zipper Lines
Purpose: pull closed bottom of net	Purpose: gather net to make it more compact for backdown	Purpose: divide net and large catch into two parts for easier handling
Description of different parts of the cable from one end to the other.		
1. Point on seiner with specified position and pursing (cable) speed.	1. Point with specified position and speed.	1. Point with either independently determined position and speed or attached to a point on net.
2. Free hanging section of cable running from seiner to a ring on the net.	2. Free hanging section of cable from the specified position to a ring on the net.	2. Section through rings (constrained motion).
3. Section through rings (constrained motion).	3. Section through rings (constrained motion).	3. Point where line is connected to the net.
4. Free hanging section of cable running from ring on net to seiner.	4. Point where line is connected to the net.	(see figures 1 & 2)
5. Point on seiner or skiff with specified position and pursing speed.	(see figures 1, 2 & 10)	
(see figures 2 & 9)		

Figure 9. Schematic Diagram of Purse Cable.

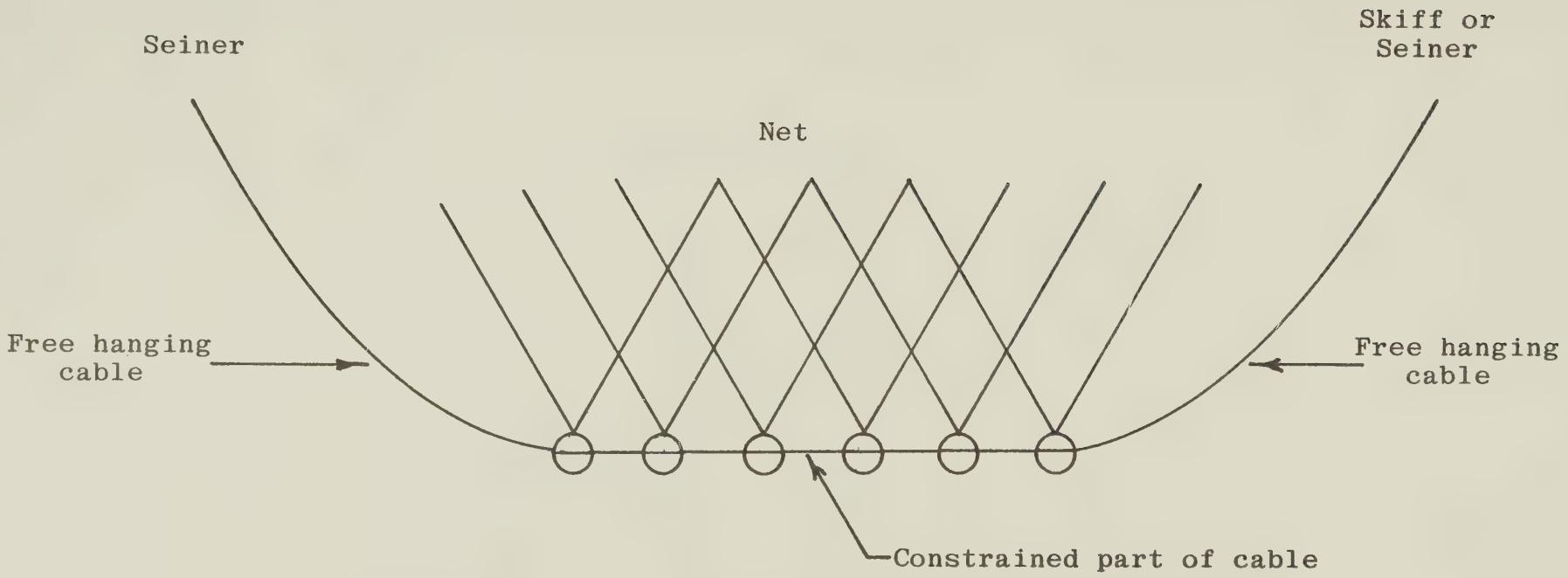
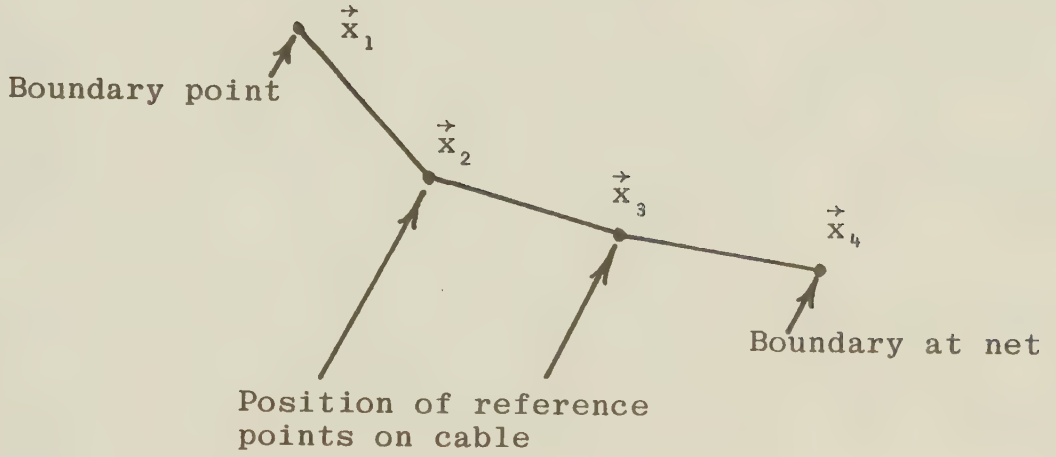


Figure 10. Schematic Diagram of Free Hanging Cable.



Schematic diagram of the model of the free hanging cable. The reference points (here $\vec{x}_2$ and $\vec{x}_3$) are free to move with the cable and along the cable. Spacing is maintained by imposing an artificial repulsion between the points. For constrained cables, these points are tied to the net and no artificial repulsion is employed.

In the following, the basic equations given in the last section are put into a form that can be solved by a computer. This requires a change from the continuous variable to a discrete variable and an explicit interface between the various parts of the problem: boundary conditions, free hanging cable, constrained cable.

5.3.1 Free Hanging Cable

The variables are the same as given in the last section. Salient characteristics of this problem are the fact that the force on the 'net' representing rings moving on the cable is zero and the mass of the net is zero. An acceleration is imposed to keep the spacing correct. Use the subscript r for these artificial rings. Then

$$\frac{d\dot{\vec{X}}_{ri}}{dt} = \hat{u}_i a_{ri} + \frac{1}{\mu_{ci}} \left\{ \vec{F}_{ci} - \hat{u}_i (\hat{u}_i \cdot \vec{F}_{ci}) - \mu_{ci} \dot{\hat{u}}_i v_{ci} \right\}$$

where the subscripts distinguish the discrete points. The relative acceleration between cable and rings is given by

$$\frac{dv_{ci}}{dt} = \hat{u}_i \cdot \left[\frac{\vec{F}_{ci}}{\mu_{ci}} - \frac{\dot{\mu}_{ci} \dot{\vec{x}}_{ci}}{\mu_{ci}} \right] - a_{ri} \, |\hat{u}_i|$$

with

$$\dot{\vec{x}}_{ci} = v_{ci} \hat{u}_i + \dot{\vec{x}}_{ri}$$

It is clear that the force, F_{ci}, and mass, μ, may be either per unit length or absolute; the units cancel. A convention must be adopted to determine the form of the quantities with subscripts. The unit vectors are chosen symmetrically; the two adjacent bars are used to calculate the drag term.

This gives

$$\vec{u}_i = \vec{x}_{ri+1} - \vec{x}_{ri-1}$$

with $$\vec{x}_{ri-1} = \vec{x}_b \text{ for } i = 1$$

$$\hat{u}_i = \frac{\vec{u}_i}{|\vec{u}_i|}$$

and $$\dot{\hat{u}}_i = \frac{1}{|u_i|}\left\{(\dot{\vec{x}}_{ri+1} - \dot{\vec{x}}_{ri-1}) - \hat{u}_i\left[\hat{u}_i \cdot (\dot{\vec{x}}_{ri+1} - \dot{\vec{x}}_{ri-1})\right]\right\}$$

The tension in the first bar is given by

$$T = T(\frac{\Delta\beta}{\Delta\alpha})$$

where $$\frac{\Delta\beta}{\Delta\alpha} = \frac{\beta_1 - \beta_b}{\alpha_1 - \alpha_b} \quad i = 1$$

or $$\left.\frac{\Delta\beta}{\Delta\alpha}\right|_i = \frac{\beta_i - \beta_{i-1}}{\alpha_i - \alpha_{i-1}}$$

At the boundary

$$\dot{\beta}_b \text{ specified}$$

$$\frac{d\beta_b}{dt} = \dot{\beta}_b$$

$$\frac{d\beta_b}{dt} = \frac{\dot{\beta}_b}{(\frac{\Delta\beta}{\Delta\alpha})}$$

where $$\left.\frac{\Delta\beta}{\Delta\alpha}\right|_b = \left.\frac{\Delta\beta}{\Delta\alpha}\right|_{i=1}$$

Note that the tension is a force that does not depend on length but on fractional extension. It must be normalized to unit length

$$\vec{F}_{Ti} = \frac{T\left(\left.\frac{\Delta\beta}{\Delta\alpha}\right|_{i+1}\right)\hat{u}_{i+1} - T\left(\left.\frac{\Delta\beta}{\Delta\alpha}\right|_{i}\right)\hat{u}_{i}}{\alpha_{i+1} - \alpha_{i}}$$

which is the tension force on the cable at i per unit length. The drag term has the same form as for the net but must be divided by the length $\alpha_{i+1} - \alpha_i$. (There is some question as to which length, α or β, is important. The Young's modulus for the materials of interest is so large that there is little difference between the two here.)

The equations of motion are given in Table 11. The equations are divided into those for integration and ancillary equations.

5.3.2 Constrained Cable

The equations of motion of the constrained part of the cable are similar to those for the unconstrained part of the cable. This is a result of choosing the equations for the unconstrained part of the cable to be compatible with those for the constrained part of the cable. The difference is that the net has a force and mass that influence the motion. Also, there is no artificial acceleration. The equations are summarized in Table 12.

Table 11. Equations Describing Free Hanging Cable.

1. Defining Equations

$\vec{x}_i$	Position in space of artificial rings. Corresponds with knot position (winch point at end).
α_i	Position of artificial ring on cable.
$\hat{u}_i = \dfrac{\vec{x}_{i+1} - \vec{x}_{i-1}}{\left\lvert \vec{x}_{i+1} - \vec{x}_{i-1} \right\rvert}$	Unit vector representing direction of cable. Becomes $\dfrac{\vec{x}_i - \vec{x}_{i-1}}{\left\lvert \vec{x}_i - \vec{x}_{i-1} \right\rvert}$ at right end, symmetrically $\dfrac{\vec{x}_{i+1} - \vec{x}_i}{\left\lvert \vec{x}_{i+1} - \vec{x}_i \right\rvert}$ at left end.
$\dot{\hat{u}}_i = \dfrac{(\dot{\vec{x}}_{ri+1} - \dot{\vec{x}}_{i-1})_\perp}{\left\lvert \vec{x}_{i+1} - \vec{x}_{i-1} \right\rvert}$	Time derivative of unit vector (rotation). Subscript $\perp$ denotes derivative perpendicular to $\hat{u}$, $\vec{v}_\perp = \vec{v} - \hat{u}(\hat{u} \cdot \vec{v})$.
$\vec{D}_i = \vec{x}_{i+1} - \vec{x}_i$	Vector between two adjacent artificial rings.
$\hat{D}_i = \dfrac{\vec{D}_i}{\left\lvert \vec{D}_i \right\rvert}$	Unit vector along which tension acts.
$D_i = \left\lvert \vec{D}_i \right\rvert$	Spacial distance between adjacent artificial rings.

Table II. Equations Describing Free Hanging Cable (cont'd).

$D\alpha_i = \alpha_{i+1} - \alpha_{i-1}$	Length of cable between rings goes to $\alpha_i - \alpha_{i-1}$ at right end, symmetrically $\alpha_{i+1} - \alpha_i$ at left end.
$\frac{\Delta s_i}{\Delta\alpha_i} = \frac{\lvert\vec{x}_{i+1} - \vec{x}_i\rvert}{\alpha_{i+1} - \alpha_i}$	Fractional stretch.
$v_{ci} = -\dot{\alpha}_i$	Velocity of cable through rings. At winch point v_{ci} is specified.
$\dot{\vec{x}}_{ci} = v_{ci}\,\hat{u}_i + \dot{\vec{x}}_{ri}$	Velocity of cable in space.
$T_i = T(\frac{\Delta s_i}{\Delta\alpha_i})$	Cable Tension as a function of stretch.
$\vec{F}_{ci} = \vec{F}_{Ti} + \mu_{ci}\vec{g} + \vec{F}_{Di}$	Tension, gravity, and drag forces per unit length on cable. μ_{ci} is mass of cable per unit length at α_i.
$\vec{F}_{Ti} = \frac{T_{i+1}\,\hat{u}_{i+1} - T_i\hat{u}_i}{D\alpha_i}$	At end points, force, $-T_{i+1}\ \hat{u}_{i+1}$, at right at $T_i\ \hat{u}_i$ at left must be added to forces at these points which are not constrained cables.
$\vec{F}_{Di} =$	Same form of drag function as for net/$(\alpha_i - \alpha_{i-1})$

Table 11. Equations Describing Free Hanging Cable (Cont'd).

$$a_{ri} = k(\alpha_{i+1} - 2\alpha_i + \alpha_{i-1}) - \sqrt{8k}\left[\alpha_i - 5(v_{ci+1} - v_{ci-1})\right]$$

Artificial acceleration to beep artificial rings properly spaced. Critically damped.

$$\vec{F}'_{ci} = \vec{F}_{ci} - \dot{\mu}_{ci}\,\vec{x}_{ci}$$

Remove momentum change due to changing cable mass.

$$\dot{\mu}_{ci} = \frac{\mu_{ci+1} - \mu_{ci-1}}{D\alpha_i}$$

Table 11. Equations Describing Free Hanging Cable.

2. Equations to be Integrated

$$\frac{d\dot{\vec{x}}_i}{dt} = \hat{u}_i\, a_{ri} - \dot{\hat{u}}_i\, v_{ci} + \frac{F_{ci\perp}}{\mu_{ci}}$$

Acceleration of artificial ring follows forces on cable perpendicular to forces on cable. $\vec{v}_\perp = \vec{v} - \hat{u}(\hat{u}\cdot\vec{v})$.

$$\frac{dv_{ci}}{dt} = \hat{u}_i \cdot \frac{F'_{ci}}{\mu_{ci}} - a_{ri}$$

Acceleration of cable relative to ring. Note that $\frac{d}{dt}(\dot{\vec{x}}_i + \hat{u}_i v_{ci}) = \frac{F_{ci}}{\mu_{ci}}$, the force equation for the cable.

$$\frac{d\vec{x}_i}{dt} = \dot{\vec{x}}_i$$

Velocity artificial ring, (ends of free cables are not free cables).

$$\frac{d\alpha_i}{dt} = -v_{ci}$$

A positive cable velocity through a ring moves the ring toward the beginning of the cable.

Table 12. Equations Describing the Constrained Cable.

1. Defining Equations

$\vec{u}_i = \vec{x}_{Ni} - \vec{x}_{Ni-1}$ $\quad (\vec{x}_{Ni-1} = \vec{x}_{rimax}$ if $i = 1$ to couple to free cable)

$\hat{u}_i = \dfrac{\vec{u}_i}{|\vec{u}_i|}$

$\dot{\hat{u}}_i = \dfrac{1}{|\vec{u}_i|} (\dot{\vec{x}}_{Ni} - \dot{\vec{x}}_{Ni-1})_{\perp}$ where $\vec{w}_{\perp} = \vec{w} - \hat{u}\,\hat{u}\cdot\vec{w}$

$\left.\dfrac{\Delta\beta}{\Delta\alpha}\right|_i = \dfrac{\beta_i - \beta_{i-1}}{\alpha_i - \alpha_{i-1}}$ Fractional stretch.

$\dot{\vec{x}}_{ci} = v_{ci}\,\hat{u}_i + \dot{\vec{x}}_{Ni}$ Cable velocity relates to relative velocity $v_{ci}\,\hat{u}_i$ and net velocity $\dot{\vec{x}}_{Ni}$.

$T_i = T(\dfrac{\Delta\beta_i}{\Delta\alpha_i})$ Tension in a function of stretch (code substitutes $\dfrac{\Delta\beta}{\Delta\alpha} \Rightarrow \dfrac{|\vec{u}|}{\Delta\alpha}$).

$\vec{F}_{ci} = \vec{F}_{Ti} + \mu_{ci}\vec{g} + \vec{F}_{Di} - \vec{f}_i$ Forces per unit length, tension gravity, drag, friction.

$\vec{F}_N =$ Forces on the part of the net attached to the cable/$(\alpha_i - \alpha_{i-1})$

μ_{Ni} = Mass of net element/$(\alpha_i - \alpha_{i-1})$.

$\vec{F}_{Ti} = \frac{T_{i+1}\hat{D}_{i+1} - T_i\hat{D}_i}{\alpha_i - \alpha_{i-1}}$ Tension becomes force per unit length.

$\vec{F}_{Di}$ = Same form of drag functions as for the net/$(\alpha_i - \alpha_{i-1})$.

$\vec{F}'_{ci} = \vec{F}_{ci} - \dot{\mu}_{ci}\dot{\vec{x}}_{ci}$ Remove momentum change due to mass charge in purse cable.

$\dot{\mu}_{ci} = \frac{\Delta\mu_{ci}}{\Delta\alpha_i}\alpha_i$

$\vec{f}_i = \text{sign}(V_{ci})\,\hat{u}_i\,\frac{\mu_{ci}\mu_{Ni}}{\mu_{ci}+\mu_{Ni}}\left|\frac{\vec{F}'_{ci}}{\mu_{ci}} - \frac{\vec{F}_{Ni}}{\mu_{Ni}}\right|_{\perp}$ Friction force add to $\vec{F}_N$, subtract from $\vec{F}'_c$.

Table 12. Equations Describing the Constrained Cable.

2. Equations to be Integrated

$$\frac{d\dot{\vec{x}}_i}{dt} = \frac{\vec{F}_{ci} + \vec{F}_{Ni}}{\mu_{ci} + \mu_{Ni}} - \frac{\mu_{ci}}{\mu_{Ni} + \mu_{ci}} \left[\dot{\hat{u}}_i \, v_{ci} + \hat{u}_i \, \hat{u}_i \cdot \left(\frac{F'_{ci}}{\mu_N} - \frac{\vec{F}_{Ni}}{\mu_{Ni}} \right) \right]$$ Acceleration of net.

$$\frac{dv_{ci}}{dt} = \hat{u}_i \cdot \left[\frac{\vec{F}'_{ci}}{ci} - \frac{\vec{F}_{Ni}}{Ni} \right]$$ Acceleration of cable relative to net.

$$\frac{d\vec{x}_i}{dt} = \dot{\vec{x}}_i$$ Velocity of net (ends of constrained cable are not constrained cables).

$$\frac{d\alpha_i}{dt} = -v_{ci}$$ A positive cable velocity through a ring moves the ring toward the beginning of the cable.

6. SUMMARY

The background literature was searched and revealed methods of mathematically formulating the equations describing the physical behavior of boat and net in the fishing process. The literature also revealed methods of converting the physical equations into numerical equations. Finally, a computational method was found that can integrate efficiently the resulting equations and a computer code that applies the method was supplied generously by Alan Hindmarsh of Lawrence Livermore Laboratory. A computer code was written to allow a user to easily set up most of the purse seining gear and process.

Due to various limitations, the code has not been exercized on large problems and, as it now stands, cannot describe the full purse seining process. However, a simulation describing a net with more than fifty macro-knots was carried out and indicated the feasibility of this magnitude of simulation. The code cannot describe the full process because the process represents a series of different structural simulations, with the geometry changing radically as different parts of the gear are attached to or disconnected from the net.

It is clear that there is more work required before a convenient simulation of the purse seining process is available. However, because there is some data on the sink rate and overall geometric behavior of the net and because the code is presently capable of easily setting up and running a large problem, the following is a reasonable recommendation. A small task should be carried out wherein

1. The SWFC gear specialists use the interactive capability of the code to describe the setting process,

2. The problem set up and simulation is carried out in consultation with the authors.

This would provide an understanding of the code's capabilities and limitations and clarify the basis for further effort.

ACKNOWLEDGEMENTS

One of the authors (T.N.D.) would like to acknowledge the help, and encouragement, of Al Stevenson of the National Marine Engineering Laboratories. It must have been through experience that he knew that the project was difficult but could yield a solid basis for further progress in net modeling. He was the only physical scientist to give encouragement. He also continually urged that the authors avoid reinventing the wheel. This lead to an extensive literature search.

The literature search produced a wealth of material on integrating stiff differential equations. The codes that applied the methods were supplied by Alan Hindmarsh of Lawrence Livermore Laboratory. Only the existance and availability of these codes made any success possible. It would be appropriate to state that the simulation described here is merely an application of the GEARS code. This cannot be overemphasized; these codes may well represent the most potent software available today.

Finally, the authors would like to acknowledge the help of the contracting officer's representative Jim Coe of the Southwest Fisheries Center. Simply said, he removed all the difficulty that is normally encountered in the contracting business.

BIBLIOGRAPHY RELATING TO NET MODELING

Anon.
"Methode zur Berechnung der Netzhulle eines pelagischen Schleppnetzes",
Fischerei-Forshung,12(2):31-33,1974.

Anon.
"The Bold Contender Purse Seine: Can It Bring Tuna Seining to the Brink of Zero Porpoise Kill?",
Fish Boat,22(1): 21-23,45,1977.

Anon.
"Die Berechnung gewichtsbehafteter Netzrotationshullen",
Fischerei-Forshung,12(2):35-38,1974.

Abe,S.
"Studies on the American Style Purse Seiner- I: Comparison with the Tuna Long Liner",
Bull. Fac. Fish. Nagasaki,43,41-44,1977.

Abe,S.
"Studies on the American Style Purse Seiner- II: Comparison with the Stern Trawler",
Bull. Fac. Fish. Nagasaki,43,45-48,1977.

Andreev, N. and Lubimov, V.
"Geometric Features of the Netting With Hexagonal Meshes",
Trudy VNIRO,122:5-40,1977.

Andreev,N.N.
"The Theory of Purse Seining",
Trudy VNIRO,41,114-123,1959.

Andreev,N.N.
"Some Problems Involved in the Designing of Purse Seines",
Trudy VNIRO,47,1962.

Andreev,N.N.
"Handbook of Fishing Gear and its Rigging",
Israel Program for Scientific Translation,Jerusalem,1966.
Jerusalem,1966.

Andreev,N.N.
"Study of the Purse Seining Process by Mechanical Imitation",
Trudy Kaliningradrybvtuza,14,1962.

Andreev,N.N. and Lubimov,V.G.
"Geometric Features of Netting with Hexagonal Meshes",
Trudy VNIRO,122,5-40,1977.

Baranov,Y.
"Construction and Use of Purse Seines",
Khabarovskoe knizhnoe izdatel'stvo,1954.

Barham,E.,Taguchi,W., and Reilly,S.
"Porpoise Rescue Methods in the Yellowfin Purse Seine Fishery and the Importance of Medina Panel Mesh Size",
Marine Fisheries Review Paper # 1246,1977.

Ben-Yami,M. and Green,R.
"Designing an Improved California Tuna Purse Seine",
Fishery Industrial Research,4(5):183-207,1968.

Blinov, V.
"Calculation fo the Coefficient of Hydrodynamic Resistence of the Net of Different Trawls",
Trudy VNIRO,97:81-85,1973.

Christensen,B.A. and Dorman,J.
"Shrimp Beam Trawls Design,Construction,and Operation",
Florida Sea Grant College Program,#6,1978.

Coe,J. and Vergne,P.
"Modified Tuna Purse Seine Net Achieves Record Low Porpoise Kill Rate",
Marine Fisheries Review Paper # 1251,1977.

Comstock,J.
"Principles of Naval Architecture",
Society of Naval Architects and Marine Engineers,N.Y.,1967.

Crewe,P.
"Trawling by Computer",
World Fishing,22(6):56,58,60,1973.

Dickson,W.
"The Use of Model Nets as a Method of Developing Trawling Gear",
Modern Fishing Gear of the World, Fishing News(Books) Ltd., London,1959.

Fischer,H.J.
"Erhohung der Effektivitat der Schleppnetzfischerei-Hauptaufgabe der Forschungs und Entwicklungstatigkeit der Abteilung Fangtechnik",
Fischerei-Forshung,16(1):23-27,1978.

Fridman,A.
"Similarity Requirements for Modeling Purse Seines",
Bureau of Commercial Fisheries,Translation #69-59000/1,1969.

Fridman,A. and Lucevic,J.
"Einige Untersuchungsergebnisse zur Modellierung von Ringwaden",
Fischerei-Forshung,12(2):39-42,1974.

Fridman,A.
"Conditions of Similarity of Fishing Gear",
Trudy KTIRP,18,131-199,1964.

Fridman,A.
"The Theory of Similarity as a Means of Designing Commercial Fishing Gear",
Trudy KTIRP,17,147-159,1964.

Fridman,A.L. and Al'tsul',B.A.
"Die Abhangigkeit der Gangtiefe des Schleppnetzes von einigen Parameter des Fangsystems",
Fischerei-Forshung,13(1):21-24,1975.

Fridman,A.
"Determination of the Conditions of Similarity of Fishing Gear and Main Trends in the Further Studies in Devising Methods of Model Construction",
Trudy KTIRP,17,129-145,1964.

Fridman,A.
"Theory and Design of Commercial Fishing Gear",
Israel Program for Scientific Translation,Jerusalem,1973.

Green,R.
"Depth-Time Sequential Analyses of the Operation of Two California Tuna Purse Seines",
Fishery Industrial REsearch,5(5):191-201,1969.

Green,R.
"Norwegian Bluefin Tuna Seines: Faster Sinking,Faster Hauling than U.S. Nets",
Pacific Fisherman,62(1):10-11,1964.

Gubenko,Y.
"Purse Seining for Pelagic Fish and Dolphins in the Black Sea",
Pishchepromizdat,1952.

Gurevic,M.I. and Fridman,A.L.
"Einige Probleme zur Theorie der Bewegung der Plagischen Schleppnetzes",
Fischerei-Forshung,12(2):11-14,1974.

Gurevic,M.I.,Fridman,A.L.,Al'tsul',B.A.,and Zagorodnij,A.V.
"Darstellung von Problemen uber die gezielte pelagische Schleppnetzfischerei
und einige Losungswege",
Fischerei-Forshung,12(2):7-10,1974.

Hamre,J.
"Some Technological Aspects of the Norwegian Tuna Purse Seine Fishery",
Fiskeridirektoratets skrifter Serie Havundersokelser,13(6):105-119,1963.

Hester,F.
"Tuna Seines: How Deep?",
Pacific Fisherman,59(12):19-20,1961.

Holts,D.B.
"The Super Apron System Construction and Installation",
S.W.F.C. Report LJ-77-30.

Hunter,J.R. et al.
"Design and Use of a Miniature Purse Seine",
Progressive Fish-Culturist,28(3):175-179,1966.

Iitaka,Y.
"Model Experiments on the Sardine Purse Seine Operating In Hyuganada",
Bull. Jap. Soc. Scient. Fish.,21(7):459-462,1955.

Imai,T. and Hirakawa,E.
"Basic Studies on the Plane Net Set into the Flowing Water-I: Comparative
Experiments on the Netting Materials",
Mem. Fac. Fish.,Kagoshima Univ.,26,157-164,1977.

Ivanov,E.P.
"Untersuchungsergebnisse zur Ausarbeitung einer analytischen Entwurfsmethode

fur Schleppnetzen",
Fischerei-Forshung,12(6):25-29,1974.

Jakobsson,J.
"Recent Developements in Icelandic Herring Purse Seining",
Modern Fishing Gear of the World(2), Fishing News(Books) Ltd., London,1964.

Kalinovskii,V.
"Purse Seining for Sardine from Small Vessels",
Trudy TNIRO,Vladivostok,1941.

Katuji,H.
"Investigation of Model Tests on Shrimp Trawl Net",
Bull. Jap. Soc. Scient. Fish.,45,153-155,1979.

Kawakami,T.
"The Theory of Designing and Testing Fishing Nets in Model",
Modern Fishing Gear of the World(2), Fishing News(Books) Ltd., London,1964.

Klust,G.
"Netting Materials for Fishing Gear",
Fishing News (Books) Ltd.,Surrey,1973.

Konagaya,T.
"Resistance of a Plane Net Set Parallel to a StReam III: The Relationship Between the Drag Coefficient and Hanging Coefficient or Angle of Incidence",
Bull. Jap. Soc. Scient. Fish.,37(11):1033-1036,1971.

Knoagaya,T.
"Studies of the Purse Seine IV: The Influence of Hanging and Net Depth",
Bull. Jap. Soc. Scient. Fish.,37(9):866-870,1971.

Konagaya,T.
"Studies of the Purse Seine III: On the Effect of Sinkers on the Performance of a Purse Seine",
Bull. Jap. Soc. Scient. Fish.,37(9):861-865,1971.

Koritzky,H.
"Form und Widerstand pelagischen Schleppnetze auf Grundlage eines Extremalprinzips",
Fischerei-Forshung,15(1):45-58,1977.

Koritzky,H.
"Widerstand und Auftrieb ebener weitmaschiger Netztuchern",
Fischerei-Forshung,12(2):51-56,1974.

Koritzky,H.
"Neue Methoden und Ergebnisse zum Bestimmen von Stromungskraften an ebenen rechteckformigen Netztielen und Netzfaden",
Fischerei-Forshung,15(2):43-47,1977.

Kristjonsson,H.
Modern Fishing Gear of the World(3), Fishing News(Books) Ltd., London,1971.

Kruse,W.
"Beitrag zur Ermittlung von Widerstand und Auftrieb ebener Netztucher im kleinen Ansteilwinkelbereich",
Fischerei-Forshung,15(2):31-41,1977.

Leitzke,H.
"Einige Betrachtungen uber das Gleichgewicht der Netzmasche im unsymmetrisch belasten Netztuch",
Fischerei-Forshung,12(1):53-56,1974.

Margetts,A.R.
"Modern Developements of Fishing Gear",in
Sea Fisheries Research,Jones,F.,ed.,John Wiley and Sons,New York,1974.

Matuda,K. and Sannomiza,N.
"Theory and Design if Bottom Drift Net III:Analytical Solutions of Equation of Motion of Gear",
Bull. Jap. Soc. Scient. Fish.,44,7-13,1978.

Matuda,K. and Sannomiza,N.
"Theory and Design of Bottom Drift Net II:Results of Numerical Analysis on the Motion of Gear",
Bull. Jap. Soc. Scient. Fish.,43,679-687,1977.

Matuda,K. and Sannomiza,N.
"Theory and Design of Bottom Drift Net I:A Method of NUmerical Analysis on the Motion of Gear",
Bull. Jap. Soc. Scient. Fish.,43,669-678,1977.

McNeely,R. and Holts,D.
"Methods of Reducing Porpoise Mortality in the Yellowfin Tuna Purse Seine Fishery",
S.W.F.C.Report,LJ-77-13.

Mc Neely,R.
"Purse Seine Revolution in Tuna Fishing",

Pacific Fisherman,59(7):27-58,1961.

Matuda,K. and Kawakami,T.
"Experimental Verification for Similarity Law on Distorted Model Net",
Bull. Jap. Soc. Scient. Fish.,34(1):23-28,1968.

Matuda,K. and Sannomiya,N.
"Theory and Design of Bottom Drift Net II: REsults of Numerical Analysis on the Motion of Gear",
Bull. Jap. Soc. Scient. Fish.,43(6):679-687,1977.

Matuda,K. and Sannomiya,N.
"Theory and Design of Bottom Drift Net I: A Method of Numerical Analysis on the Motion of Gear",
Bull. Jap. Soc. Scient. Fish.,43(6):669-678,1977.

Mouri,K. and Higo,N.
"On an Approximation Equation Applicable to the Designing of the Trawl Net I: On the Net Mouth Central Height of the Four Panelled Net",
Mem. Fac. Fish.,Kagoshima Univ.,25,107-115,1976.

Nomura,M. et. al.
"Study of the Behavoir of the Purse Seine IV: Large Scale Model Experiment of Horse Mackerel and Mackerel Purse Seine",
Tokai Regional Fisheries Research Lab. Bull.,49,63-80,1967.

Okubo,A.,Sakamoto,W.,Inagaki,T.,and Kuroki,T.
"Studies on the Schooling Behavior of Fish V: Note on the Dynamics of Fish Schooling",
Bull. Jap. Soc. Scient. Fish.,43(12):1369-1377,1977.

Orbach,M.
"Hunters,Seamen and Entrepreneurs (The Tuna Seinermen of San Diego)",
University of California Press,1978.

Rothbarth,H.
"Streckenmessung am Schleppnetzgespchirr Einsatz eines hydroakustischen Artwortgeber",
Fischerei-Forshung,12(1):,57-60,1974.

San,N. and Fuwa,S.
"Comparison of the Gear Efficiency of two Types of Trawl Net",
Mem. Fac. Fish.,Kagoshima Univ.,24,47-56,1975.

Sanders,M. and Morgan,A.
"Fishing Power,Fishing Effort,Fishing Density,Fishing Intensity,and Fishing Mortality",
J. Cons. Int. Explor. Mer.,37,36-40,1976.

Saunders,H.
"Hydrodynamics in Ship Design",
Society of Naval Architects and Marine Engineers,N.Y.,1965.

Schmidt,P.
"The Puretic Power Block and its Effect on Modern Purse Seining",
Modern Fishing Gear of the World, Fishing News(Books) Ltd., London,1959.

Scharping,K.
"Beitrag zur Untersuchung der Stromungsverhaltnisse am Fischereinetz",
Fischerei-Forshung,12(2):57-63,1974.

Schumacher,W.
"Untersuchung zum statischen und dynamischen Verhalten des Fanggerates Schleppnetz unter besonderer Beruchsichtigung der Scherkorper",
Fischerei-Forshung,13(1):31-39,1975.

Schumacher,W.
"Beitrag zum dynamischen Verhalten von Scherkorpern",
Fischerei-Forshung,12(2):65-71,1974.

Stengel,H. and Hahlbeck,W.
"Arbeitsdiagramme zur Berechnung schwerer stromungsbelasteter Faden",
Fischerei-Forshung,12(2):43-49,1974.

Stengel,H. and Leitzke,H.
"Beitrag zum hydrodynamischen Widerstand und Auftrieb ebener Netztuchen",
Fischerei-Forshung,13(1):41-48,1975.

Stengel,K. and Kropelin,J.
"Berucksichtigung der Dehnung bei der Berechnung ger Zugkraft und
Durchhangsform von Schlepptrossen bei gleichformiger Bewegung in
einem flussigen Medium",
Fischerei-Forshung,13(1):49-55,1975.

Stengel,H.
"Beitrag zur Theorie und Berechnung pelagischen Zweischiffsschleppennetze",
Fischerei-Forshung,15(2):23-29,1977.

Takeshi,Y.
"On the Motion of Fishing Gear Against Wave Action II: Vertical Motion of
a Float Exposed to Sine Wave",
Bull. Jap. Soc. Scient. Fish.,45,335-339,1979.

Thomson,D.
"The North American and European Purse Seine Types Compared",
World Fishing,16(6):73-79,1967.

Vinogradov,N.
"The Sinking Speed of Purse Seines' Leadlines",
Trudy AzCherNIRO,14,23-28,1950.

Vinogradov,N.
"Herring Purse Seining",
Petrozavodsk,1955.

Yoza,K. et. al.
"Studies on the Characteristics of Leader of Setnet II: Length and
Direction of the Leader",
Bull. Jap. Soc. Scient. Fish.,43,1387-1390,1977.

Zagorodnij,A. V.
"Untersuchung des Steuervorganges plagischerr Schleppnetze mittels Modellen",
Fischerei-Forshung,12(2):15-19,1974.

Zagorodnij,A.
"Ergebnisse der Modellerprobung einer Mapstabssenie eines pelagischen
Schleppnetzes",
Fischerei-Forshung,13(1):25-29,1975.

BIBLIOGRAPHY RELATING TO MATHEMATICAL METHODS

Alder, B., Fembach, S., and Rotenberg, M. (editors)
"Methods in Computational Physics,"
Vol. 9, Academic Press, New York, 1970.

Bazant, Z. P.
"Spurious Reflection of Elastic Waves in Nonuniform Finite Element Grids,"
Comp. Meth. App. Mech. Eng. 16:19-100,1978.

Byrne, G. D. and Hindmarsh, A. C.
"A Polyalgorithm for the Numerical Solution of Ordinary Differential Equations,"
ACM Trans. on Math. Software, 1:71-96,1975.

Carver, M. B.
"Efficient Integration Over Discontinuities in Ordinary Differential Equation Simulations,"
Math. Comp. Sim. 20:190-196,1978.

Carver, M. B. and Mac Ewen, S. R.
"Numerical Analysis of a System Described by Implicity - Defined Ordinary Differential Equations Containing Numerous Discontinuities,"
Appl. Math. Modeling, 2:280-286,1978.

Carver, M. B., Stewart, D. G., Blair, J. M. and Selander, W. N.
"The Forsim VI Simulation Package for the Automated Solution of Arbitrary Defined Partial and/or Ordinary Equation Systems,"
Atomic Energy of Canada Limited Report AECL-5821, 1978.

Chan, Y. N. I., Birnbaum, I. and Lapidus, L.
"Solution of Stiff Differential Equations and the Use of Imbedding Techniques,"
Ind. Eng. Chem. Fundam. 17:133-148,1978.

Ciment, M., Leventhal, S. and Weinberg, B.
"The Operator Compact Implicit Method for Parabolic Equations,"
report NSWC/WOL/TR 77-29, Naval Surface Weapons Center, White Oak Laboratory, Silver Spring, Maryland.

Concus, P., Golub, G. H. and O'Leary, D. P.
"Numerical Solution of Nonlinear Elliptic Partial Differential Equations by a Generalized Conjugate Gradient Method,"
Computing 19:321-339,1978.

Enright, W.
"On the Efficient and Reliable Numerical Solution of Large Linear System of ODES,"
Proceedings of the 1978 IEEE Conference on Decision and Control Including the 17th Symposium on Adaptive Processes, 54-59, 1979.

Felippa, C. A.
"Incremental Finite Element Matricies,"
(Comments on paper by S. Rajasebaran and D. Murray)
J. Structural Division ASCE, 100:2521-2523, 1974.

Felippa, C. A.
"Optimization of Finite Element Grids by Direct Energy Search,"
Appl. Math. Modeling, 1:93-96, 1976.

Felippa, C. A.
"Constraint Definition Techniques in Solution of Discrete Boundary-Value Problems,"
SIAM Review, 18:802-803,1976.

Felippa, C. A.
"Numerical Experiments in Finite Element Grid Optimization by Direct Energy Search,"
Appl. Math. Modeling, 1:239-244,1977.

Felippa, C. A.
"Error Analysis of Penalty Function Techniques for Constrained Definition in Linear Algebraic Systems,"
Int. J. Numerical Methods in Eng., 11:709-728,1977.

Felippa, C. A.
"Local Iterative Procedures for Improving Penalty Function Solutions of Finite-Element Systems,"
SIAM Review, 20:624,1978.

Gear, C. W.
"Ordinary Differential Equation Techniques for Partial Differential Equations," in "Formulations and Computational Algorithms in Finite Element Analysis," eds. K. J. Bathe, J. T. Oden, W. Wunderlich, M.I.T., 1977.

Helliwell, W. S.
"A Fast Implicit Iterative Numerical Method for Solving Multidimensional Partial Differential Equations,"
AIAA Journal 16:663-666,1978.

Hopkins, T. R. and Wait, R.
"A Comparison of Galerkin, Collocation and the Methods of Lines for Partial Differential Equations,"
Int. J. Num. Meth. Eng., 12:1081-1107,1978.

Houstis, E. N., Lynch, R. E. and Rice, J. R.
"Evaluation of Numerical Methods for Elliptic Partial Differential Equations,"
J. Comp. Phys., 27:323-350,1978.

Jocabs, D. (editor)
"The State of the Art in Numerical Analysis,"
Academic Press, London 1977.

Kamel, H. A. and McCabe, M. W.
"Direct Numerical Solution of Large Sets of Simultaneous Equations,"
Computers and Structures, 9:133-123,1978.

Kershaw, D. S.
"The Incomplete Cholesky-Conjugate Gradient Method for the Iterative Solution of Systems of Linear Equations,"
J. Comp. Phys., 26:43-65,1978.

Kreiss, H. O.
"Problems with Different Time Scales,"
Numerical Analysis, 95-106,1978.

Kreiss, H. O. and Oliger, J.
"Methods for the Approximate Solution of Time Dependent Problems,"
Global Atmospheric Research Programme (GARP) Publications Series No. 10, 1973.

Monkar, D. P. and Powell, G. H.
"Evaluation of Solution Schemes for Nonlinear Structures,"
Computers & Structures, 9:223-236,1978.

Potter, D.
"Computational Physics,"
John Wiley & Sons, New York, 1973.

Proskurowski, W. and Widlund, O.
"On the Numerical Solution of Helmholtz's Equation by the Capacitance Matrix Method,"
Math. of Comp. 30:433-468,1976.

Scaife, B. K. P. (editor)
"The State of the Art in Numerical Analysis,"
Academic Press. London, 1976.

Schoenstadt, A. L.
"A Transfer Function Analysis of Numerical Schemes Used to Simulate Geostrophic Adjustment,"
report NPS-53-79-001, Naval Postgraduate School, Monterey, California, 1978.

Schumann, U.
"A Direct Method for the Solution of Poisson's Equation with Neumann Boundary Conditions on a Staggered Grid of Arbitrary Size,"
J. Comp. Phys., 20:171-182,1976.

Shampine, L. F. and Gear, C. W.
"A User's View of Solving Stiff Ordinary Differential Equations,"
SIAM Review, 21:1-17,1979.

Sincovec, R. F. and Madsen, N. K.
"Software for Nonlinear Partial Differential Equations,"
Software, 1:232-260,1975.

Sincovec, R. F. and Madsen, N. K.
"Algorithm 494, PDEØNE, Solutions of Systems of Partial Differential Equations,"
ACM Trans. Math. Software, 1:261-263,1975.

BIBLIOGRAPHY RELATING TO CABLE METHODS

Baumeister, Theodore, Editor
"Standard Handbook for Mechanical Engineers,"
Seventh edition, 1967.

DeZoysa, A. P. K.
"Steady-State Analysis of Undersea Cables,"
Ocean Engng., 5:203-223,1978.

Felippa, C. A.
"Finite Element Analysis of Three-Dimensional
Cable Structures,"
311-324 in "Computational Methods in Nonlinear
Mechanics," ed. J. T. Oden et al., 1975,
Texas Institute for Computational Mechanics.

Foster, E. P., and Beaufait, F. W.
"Analysis of Cable Net-Panel Roof System,"
Computers and Structures 9:501-521,1978.

Henderson, J. F.
"Some Towing Problems with Faired Cables,"
Ocean Engng. 5:105-125,1978.

Hood, C. G.
"A General Stiffness Method for the Solution of
Nonlinear Cable Networks with Arbitrary Loading,"
Computers and Structures 6:391-396,1976.

Morris, N. F., and Birnstiel, C.
"Dynamic Response of Three-Dimensional Cable Systems,"
NTIS document PB-235, 853, 1974.

Park, K. C., and Felippa, C. A.
"Computational Aspects of Time Integration Procedures
in Structural Mechanics, Part 2: Error Propagation,"
Journal of Applied Mechanics Transactions of the ASME
Series E 43:603-611,1978.

Pietrzak, J.
"Matrix Formulation of Static Analysis of Cable Structures,"
Computers and Structures, 9:39-42,1978.

Sangeter, K. G., and Batchelor, B.
"Nonlinear Dynamic Analysis of 3-D Cable Networks,"
301-310 in Computational Methods in Nonlinear Mechanics,
ed J. T. Oden et al., 1975, Texas Institue for Computational
Mechanics.

Van Fossen, D. B.
"FESAP--Design Program for Static and Dynamic Structural Analysis,"
Computers and Structures 9:371-376,1978.

Wang, H. T.
"Determination of the Accuracy of Segmented Representations
of Cable Shape,"
ASME Ser. B 97:472-478,1975.

Wingham, P. J., and Keshavan, N. R.
"Predicting the Equilibrium Depths of a Body Towed by a
Faired Cable,"
Ocean Engng. 5:15-35,1978.

Winget, J. M., and Huston, R. L.
"Cable Dynamics - A Finite Segment Approach,"
Computers and Structures 6:475-480,1976.

APPENDIX

Program Parameters

Some parameters are not available to the user for modification in the input stream. These may be changed by changing their values in the code. The coefficient of friction, SIGMA = .2, between the constrained cables and the rings was chosen to be small since the rings slide readily over the cables. Other values are for the ship and are partially from Abe (1977) or estimated as shown in the table. The values are set in the subroutine FSANER.

Table 13. Seiner Parameters.

Parameter	Value	Physical Meaning/Method Used to Estimate if *
SLØNG	50m	seiner length
SDEEP	3m	seiner mean draft
SMASS	1×10^6 kg	seiner mass
(beam)	11m	used in estimating other parameters
(average height)	5m	used in estimating other parameters
APX*	55 m^2	area as seen by wind from forward direction = average height x beam
APY*	250 m^2	area as seen by wind from side direction = average height x SLØNG
XCA*	3 m	center of area forward of center of mass
SWET*	572 m^2	wetted area ≈ SLØNG * BEAM $*\sqrt{2}$
SINRTZ	1×10^8 kgm^2	seiner moment of inertia ≈ $(.2*SLØNG)^2*SMASS$

DEVELOPMENT OF A NUMERICAL SIMULATION OF PURSE SEINE FISHING

II. Extension of the Numerical Simulation of the Purse Seine Fishing Process

TABLE OF CONTENTS

		Page
Section 1:	INTRODUCTION	1-1
Section 2:	MODULARIZATION OF THE SIMULATION	2-1
Section 3:	STRUCTURE CHANGE	3-1
Section 4:	IMPROVED TREATMENT OF RINGS	4-1
	4.1 EQUATIONS OF MOTION FOR THE CONSTRAINED CABLE	4-1
	4.2 ILLUSTRATION OF PROPER RING FUNCTIONING	4-9
Section 5:	SUMMARY	5-1

LIST OF FIGURES

FIGURE 1. Old SIMUL8 Program Subroutine Hierarchy

FIGURE 2. SIMUL8 Program Subroutine Hierarchy

FIGURE 3. SIMUL8 Subroutine Summary

FIGURE 4. SIMUL8 Data Specifications

FIGURE 5. NET Program Subroutine Hierarchy

FIGURE 6. NET Subroutine Summary

FIGURE 7. NET Data Specifications

FIGURE 8. RINGS Net: Creation Commands and Schematic

FIGURE 9. "Rings Up" Scenario Definition Commands

FIGURE 10. "Rings Up" Simulation Time Evolution

FIGURE 11. Purse Cable and Rings Schematic

FIGURE 12. Ring Functioning: Scene Definition and Time Evolution

LIST OF TABLES

TABLE 1(a). DEFINITIONS USED IN THE EQUATIONS OF MOTION OF THE RINGS

TABLE 1(b). ABBREVIATIONS USED FOR i^{th} RING

Section 1

INTRODUCTION

The primary goal of the current project was to modularize the simulation code. Previously, the differential equations describing the components in the purse seining process were located in a single subroutine. Not only did this make the code difficult to comprehend, but it did not allow the flexibility required to:

- add or delete components during execution,
- modify structural connectivity during execution, and
- add calculations of biologically important variables.

These features are essential to the simulation of the purse seine fishing process. However, this flexibility is not easily attained; it amounts to solving a set of coupled differential equations whose coupling, number, and type vary as a function of the solution in an extremely general fashion.

The solution to such a problem had not been previously attempted.

Another relatively minor, but still important, feature that required redressing was a fatal quirk in the description of the motion of rings on cables. In the previous simulation, rings could change order on the cable and the calculation would stop. This problem was solved.

This report has three main sections addressing the following:

(1) modularization,

(2) structure changes, and

(3) improved ring treatment.

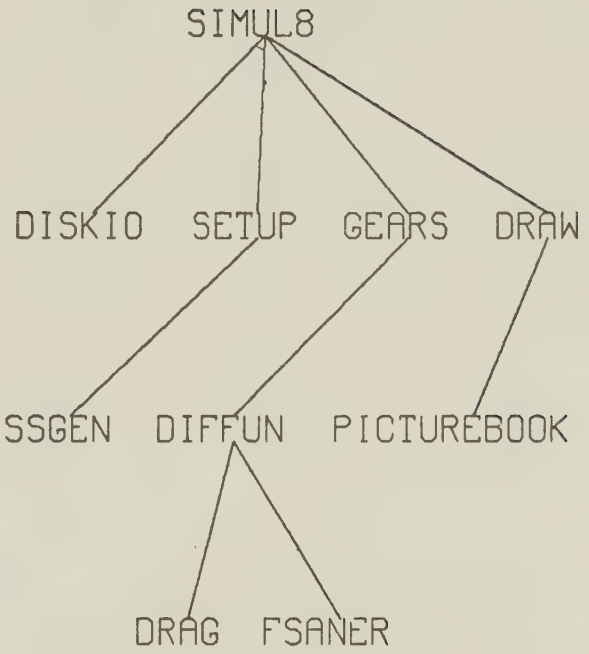
Allowing components to be treated as modules is a prerequisite for handling structural changes and is presented in one section. By specifying the system in a mathematical "graph" as nodes and connections, the possibility of describing structure changes as modifications of the graph (along with proper initialization) arises. The method of taking advantage of this possiblity is described in a subsequent section. Finally, an improved method of treating rings on a cable is described that was designed to keep the rings from crossing and, additionally, avoid some difficulties that arose earlier when the cable passing through a ring changed direction radically.

Section 2

MODULARIZATION OF THE SIMULATION

The major goal of the present project was to redesign the simulation program in a more generalized, modular fashion. The original version of this program did very little toward separation of function of the generic components out of which the problem is built. The result was a poorly organized program in which the equations of motion of all the various components were confined to one long subroutine: DIFFUN. This method of organization was manageable during the early stages of the project, but became less and less tractable as more pieces were added. Finally, it was apparent that the logic flow of the code could only be comprehended with great difficulty, and that no more components could be added to the simulation in a reasonable way. One particular manifestation of this overloaded design was the appearance of a "bug" wherein rings that slid along cables tended to cross over one another and stop the program. (This discussed in detail in a later section.) A simple solution to this was conceived, but an implementation of the fix was hindered due to the complex logic that presented itself. Figure 1 shows the overall structure of the original version of the SIMUL8 program. The relative simplicity of this diagram belies the bewildering internal flow structure, especially of DIFFUN.

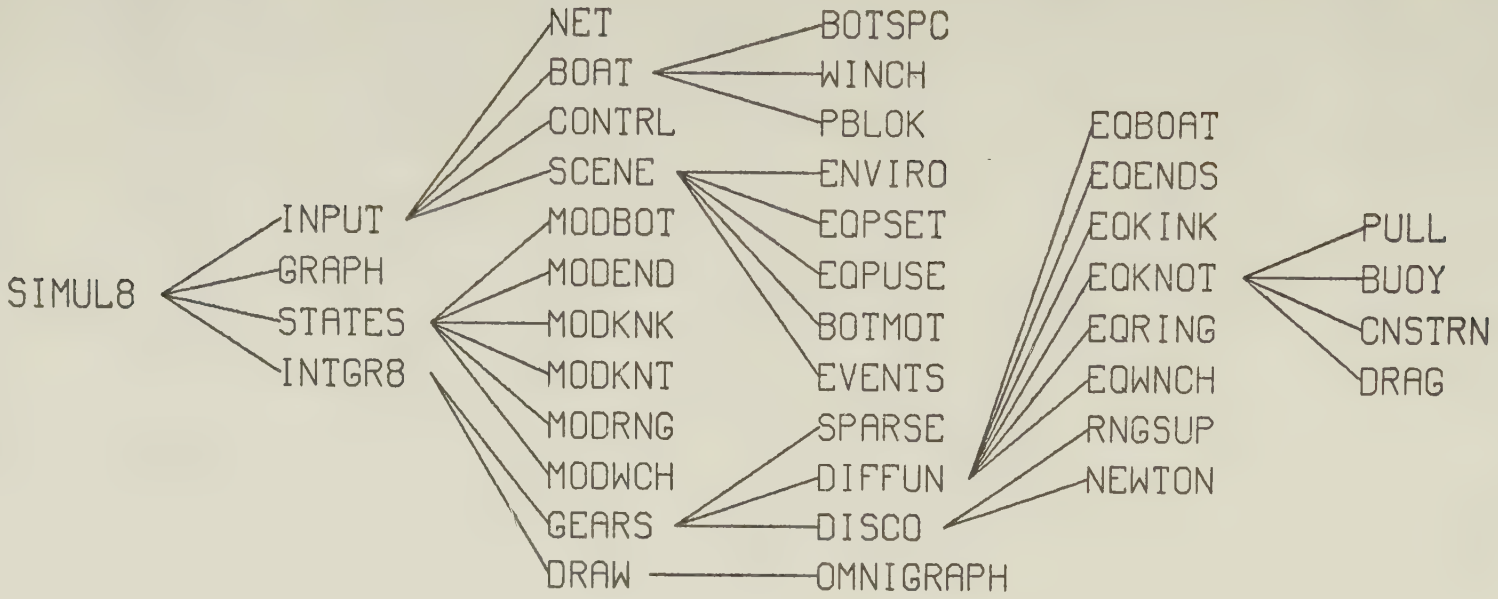
Figure 2 shows the organization of the present version of SIMUL8. A closer look at this tree diagram reveals the modularization strategy for this program: the routines ADDKNT, GETKNT, MODKNT, and EQKNOT all refer to the generic component known as a "knot". Respectively, they store and access the attributes of, and set-up and define the equations for, each separate "knot".



UTILITIES

DOT, VNET, VMAG, WATERV, XINTRP

FIGURE 1. Old SIMUL8 Program Subroutine Hierarchy



UTILITIES

DATA ACCESS/MANAGEMENT

ADDBLK, ADDCBL, ADDEND, ADDKNK, ADDKNT
ADDLNK, ADDNOD, ADDRNG, ADDTAG, ADDTWN
GETBLK, GETCBL, GETEND, GETKNK, GETKNT
GETLNK, GETNOD, GETRNG, GETTAG, GETTWN
FNDLNK, FNDNOD, FNDTAG, COMPNT, KOMPNT
PACK2, UNPAK2, PACK4, UNPAK4, LENMOD, LOCMOD
COLAPS, BUMP, KILL, TAGTYP, ACTIV8

MISCELLANEOUS

DOT, UNITV, VMAG, WATERV, XINTRP
GETTBL, REFRNC, TBLDOT, XYZ

FIGURE 2. SIMUL8 Program Subroutine Hierarchy

Versions of these four basic routines exist for other components as well, such as "Ends", "Rings", "Kinks", etc. Thus, the addition of newly defined components to the system proceeds in a much more straightforward fashion than was the case originally. The sophistication of the simulation should now grow in an organized way. Moreover, the various components will not interfere with one another, in a global sense, due to their isolation in separate routines and data sets.

The master organizing principal around which the entire simulation is defined is the "graph". This mathematical construct is simply a way of describing a set of objects that are interconnected in a general fashion. In the present case, the nodes of the graph represent those components for which equations of motion are defined: knots, rings, kinks, ends, winches, the boat, etc. The links of the graph represent the interconnections themselves, which at present are either cable elements or twines. The utility of this notion of a graph is that it is an abstract concept capable of organizing the entire problem; and yet it may be manipulated independently of the details of each and every node and link. It is a very powerful and general foundation upon which to build this complex simulation.

Figure 3 presents an alphabetized summary of the subroutines composing SIMUL8. The functions of the routines shown in the tree diagram of Figure 2 are described here. Figure 4 gives the data structure specifications for SIMUL8. Figures 2, 3, and 4 serve as a complete guide, short of a code listing, to the SIMUL8 program.

Figures 5, 6, and 7 are the corresponding tree diagram, subroutine summary, and data specification for the NET program, included here for completeness.

SUBROUTINE ACTIV8(TYPE,IPOINT,ACTIV)

Specify the ACTIVity of a 'NODE' or 'LINK' (TYPE) of the graph. ACTIV may be 'DEAD'/'LIVE' IPOINT points to the item being changed.

SUBROUTINE ADDBLK(POSITN,NAME,IBLK)

Add a 'BLOK' type component to the list of basic components. The data describing power blocks, from the argument list, will reside in /PBLOCK/.

SUBROUTINE ADDCBL(TENSHN,DRAG,TYPLFT,NUMLFT,TYPRGT,NUMRGT,NAME,
* IDENTB,ICBL)

Add a 'CABL' type component to the list of basic components. The data describing cables, from the argument list, will reside in /CABLES/.

SUBROUTINE ADDEND(WEIGHT,VOLUME,NAME,ISEQNC,IEND)

Add a 'ENDS' type component to the list of basic components. The data describing ends, from the argument list, will reside in /ENDS/.

SUBROUTINE ADDKNK(NAME,ISEQNC,IKNK)

Add a 'KINK' type component to the list of basic components. The data describing kinks, from the argument list, will reside in /KINKS/.

SUBROUTINE ADDKNT(WEIGHT,VOLUME,IKNT)

Add a 'KNOT' type component to the list of basic components. The data describing knots, from the argument list, will reside in /KNOTS/.

SUBROUTINE ADDLNK(TYPE,NUMBER,ACTIVE,TYPLFT,NUMLFT,TYPRGT,NUMRGT,
* NNAME,NAMES)

Define a link in the graph of the problem. The TYPE and serial NUMBER of the linking component are specified. The link may be ACTIVE or inactive ('LIVE'/'DEAD'). It may also have NNAME logical NAMES associated with it. The type and serial numbers of the left and right connected components are specified in TYPLFT,NUMLFT, and TYPRGT, NUMRGT.

SUBROUTINE ADDNOD(TYPE,NUMBER,ACTIVE,NNAME,NAMES)

Define a node in the graph of the problem. The TYPE and serial NUMBER of the node component are specified. The node may be ACTIVE or inactive ('LIVE'/'DEAD'). It may also have NNAME logical NAMES associated with it.

SUBROUTINE ADDRNG(WEIGHT,VOLUME,NAME,ISEQNC,IRNG)

Add a 'RING' type component to the list of basic components. The data describing rings, from the argument list, will reside in /RINGS/.

SUBROUTINE ADDTAG(TYPE,KK,NN,MM)

Find room in the /TAGS/ array to store another TAG variable.

FIGURE 3. SIMUL8 Subroutine Summary

SUBROUTINE ADDTWN(TENSHN,DRAG,TYPLFT,NUMLFT,TYPRGT,NUMRGT,BARLEN,
* ITWN)

Add a 'TWIN' type component to the list of basic components. The data describing twines, from the argument list, will reside in /TWINES/.

SUBROUTINE BOAT

Oversee all program input concerning the seiner static description.

SUBROUTINE BOTMOT

Process program input, under the SCENE heading, describing seiner motion during the simulation. The data will reside in /SEINER/.

SUBROUTINE BOTSPC

Decode program input, under the SEINER heading, describing the seiner physical specifications. The data will reside in /SEINER/.

SUBROUTINE BUMP(MODE,ICOUNT,IBUMP)

This routine protects the PARAM and KONECT arrays from being overfilled. If MODE='PARM' or 'KNCT', the corresponding array counter will be BUMPed by IBUMP and checked against limits. If MODE='INIT', both array counters will be zeroed. An auxiliary counter, ICOUNT, will also be incremented by IBUMP, if desired.

FUNCTION BUOY(INFO,IPOINT,Z)

If INFO contains the proper keywords, then a buoy is attached to the module that called this routine. Return the displaced volume of the BUOY at depth Z. IPOINT points into INFO and is bumped as INFO is being used.

SUBROUTINE CNSTRN(TIME,INFO,IPOINT,VELCTY,ACCEL)

If INFO contains the proper keywords, impose motion constraints on the module that called this routine: set the VELCTY & ACCELeration. IPOINT points into INFO and is bumped as INFO is used.

SUBROUTINE COLAPS(MODE,ARRAY,N)

Collapse an ARRAY of N elements. If MODE='FULL', a complete sweep will take place through the array, removing all flaged values. N will be reset accordingly. If MODE='PART', only the last elements of the ARRAY that are flaged in an uninterupted sequence will be removed.
If MODE='TAGS', the complex TAG array will be fully collapsed.

FUNCTION COMPNT(I)

The value of COMPNT is the four character mnemonic that identifies component type I. The component types are:

I	MNEMONIC
1	'KNOT'
2	'RING'

FIGURE 3. SIMUL8 Subroutine Summary
(Continued)

```
3     'KINK'
4     'ENDS'
5     'TWIN'
6     'CABL'
7     'WNCH'
8     'BLOK'
9     'BOAT'
```

```
    SUBROUTINE CONTRL
```

Process input commands that control the simulation solution.
The data will reside in /CNTROL/.

```
    SUBROUTINE DIFFUN(N,TIME,SYSTEM,DSYSDT)
```

The SYSTEM is sectioned into "modules", each of which represents an active node in the system graph. For each module, identify its component type and form the derivatives of its SYSTEM variables, and return them in DSYSDT.

```
SUBROUTINE DISCO(Y,N0,TIME,H,IDISCO)
```

Disco looks for changes (discontinuities) between the last two accepted steps, and the last accepted and a predicted next step. If a siginificant change occured between the last two steps, flags are set and control is returned. If no siginificant change occured, such a change is searched for in the next time step. If found, the time to it is estimated. If the time step is small enough, a predictor hop is made (otherwise the time step is reset and control returned). If the hop is made, then siginificant change is implied, and so flags are set, and control is returned. Communication is as follows:

Y	The Nordsieck array, N0 by LMAX.
TIME	Current time.
H	Time step.

Ouptuts:

H	New recommended time step, when IDISCO > 1
IDISCO	1=no changes 2=Recommend a new time step, as when moving to a discontinuity,or restarting after one. 3=Recommend a new time step and restart, as after a discontinuity, when a new Jacobian should be calculated and old derivatives are useless. 4=Recommend a new time step and restructure equations, as when the equation set and Jacobian structure change.

```
    FUNCTION DOT(V1,V2)
```

Calculate the scalar (dot) product of two vectors

```
    SUBROUTINE DRAG(DLIFT,DMINUM,DFLOW,VELCTY,VWATER,BLENTH,
   * UNIT,DRAGG)
```

Calculate the DRAGG force on a single bar. The bar of length BLENTH, orientation UNIT, and given VELCTY, is exposed to water velocity VWATER. Three unitless drag parameters, to specify the lift, flow, and minimum drag effects that are characteristic of this bar, are passed as DLIFT, DFLOW, and DMINUM, respectively.

FIGURE 3. SIMUL8 Subroutine Summary
(Continued)

```
D=p/2*V*|V|*cd*a+p/2*|V|**2*c1*a* VUx(VUxU)/|VUx(VUxU)|

   where V=relative velocity of water to knot=VWATER-VELCTY
        |V|=relative speed of knot=SPEED
         VU=unit vector in V direction
          U=unit vector in direction of bar or cable element=UNIT
          p=water density=1000KG/M**3
          a=projected area of a bar (length x diameter)
          cd=c2+c3 sin**3(gam)
          c1=c1 sin**2(gam) cos(gam)  where cos(gam)=U.VU
          c1,c2,c3=experimental parameters (unitless)

Here, [DLIFT,DMINUM,DFLOW]=a p/2 c[1,2,3] scale**2 where scale=1
for cables or = linear scale factor for net elements.


    SUBROUTINE ENVIRO

Process input commands, under the SCENE heading, that describe the
physical environment. Data will reside in /ENVRON/


    SUBROUTINE EQBOAT(TIME,VARIBL,SYSTEM,PARAM,KONECT,DERIV)

This routine contains the equations for the DERIVatives of the system
VARIBLs that are associated with the seiner, or "BOAT".
The BOAT requires 6 state variables:
           VARIBL(1)= X position
           VARIBL(2)= Y position
           VARIBL(3)= U velocity (along boat axis)
           VARIBL(4)= V velocity (transverse to boat axis)
           VARIBL(5)= THETA angle of boat heading w.r.t. X axis.
           VARIBL(6)= THEDOT angular velocity of boat heading.
Given the values of the above VARIBLs at the present TIME, as well as
the current values of all the other SYSTEM variables, calculate the
time DERIVatives of the state variables. The
information required to compute these derivatives for the boat is
shared between the PARAM and KONECT arrays. PARAM contains all the
physical parameters for the present component, stored in a predeter-
mined order (see MODBOT); likewise, KONECT has all the connection data
describing links to other SYSTEM components.


    SUBROUTINE EQENDS(TIME,VARIBL,SYSTEM,PARAM,KONECT,DERIV)

This routine contains the equations for the DERIVatives of the system
VARIBLs that are associated with "ENDS". ENDS are defined as those
nodes having one cable bar attached.
Each END requires 7 state variables:
           VARIBL(1)= X position
           VARIBL(2)= Y position
           VARIBL(3)= Z position
           VARIBL(4)= X velocity
           VARIBL(5)= Y velocity
           VARIBL(6)= Z velocity
           VARIBL(7)= cable reference position (ALPHA)
Given the values of the above VARIBLs at the present TIME, as well as
the current values of all the other SYSTEM variables, calculate the
time DERIVatives of the state variables for a particular END. The
information required to compute these derivatives for a given END is
shared between the PARAM and KONECT arrays. PARAM contains all the
physical parameters for the present component, stored in a predeter-
mined order (see MODEND); likewise, KONECT has all the connection data
describing links to other SYSTEM components.
```

FIGURE 3. SIMUL8 Subroutine Summary
(Continued)

SUBROUTINE EQKINK(TIME,VARIBL,SYSTEM,PARAM,KONECT,DERIV)

This routine contains the equations for the DERIVatives of the system VARIBLs that are associated with "KINKS". KINKS are defined as those nodes having two cable bars attached, with no twine(s).
Each KINK requires 8 state variables:

VARIBL(1)= X position
VARIBL(2)= Y position
VARIBL(3)= Z position
VARIBL(4)= X velocity
VARIBL(5)= Y velocity
VARIBL(6)= Z velocity
VARIBL(7)= cable reference position (ALPHA)
VARIBL(8)= cable reference velocity (ALPDOT)

Given the values of the above VARIBLs at the present TIME, as well as the current values of all the other SYSTEM variables, calculate the time DERIVatives of the state variables for a particular KINK. The information required to compute these derivatives for a given KINK is shared between the PARAM and KONECT arrays. PARAM contains all the physical parameters for the present component, stored in a predetermined order (see MODKINK); likewise, KONECT has all the connection data describing links to other SYSTEM components.

SUBROUTINE EQKNOT(TIME,VARIBL,SYSTEM,PARAM,KONECT,DERIV)

This routine contains the equations for the DERIVatives of the system VARIBLs that are associated with "KNOT". Knots are defined as those nodes that have twine and possibly cable ends attached.
Each KNOT requires 6 state variables:

VARIBL(1)= X position
VARIBL(2)= Y position
VARIBL(3)= Z position
VARIBL(4)= X velocity
VARIBL(5)= Y velocity
VARIBL(6)= Z velocity

Given the values of the above VARIBLs at the present TIME, as well as the current values of all the other SYSTEM variables, calculate the time DERIVatives of the state variables for a particular knot. The information required to compute these derivatives for a given knot is shared between the PARAM and KONECT arrays. PARAM contains all the physical parameters for the present component, stored in a predetermined order (see MODKNT); likewise, KONECT has all the connection data describing links to other SYSTEM components.

SUBROUTINE EQRING(TIME,VARIBL,SYSTEM,PARAM,KONECT,DERIV)

This routine contains the equations for the DERIVatives of the system VARIBLs that are associated with "RING". Rings are defined as those nodes that have twine, a running cable, and possibly cable ends attached.
Each RING requires 8 state variables:

VARIBL(1)= X position
VARIBL(2)= Y position
VARIBL(3)= Z position
VARIBL(4)= X velocity
VARIBL(5)= Y velocity
VARIBL(6)= Z velocity
VARIBL(7)= cable reference position (ALPHA)
VARIBL(8)= cable reference velocity (ALPDOT)

Given the values of the above VARIBLs at the present TIME, as well as the current values of all the other SYSTEM variables, calculate the time DERIVatives of the state variables for a particular RING. The information required to compute these derivatives for a given RING is shared between the PARAM and KONECT arrays. PARAM contains all the physical parameters for the present component, stored in a predeter-

FIGURE 3. SIMUL8 Subroutine Summary
(Continued)

mined order (see MODRNG); likewise, KONECT has all the connection data describing links to other SYSTEM components.

SUBROUTINE EQWNCH(TIME,VARIBL,SYSTEM,PARAM,KONECT,DERIV)

This routine contains the equations for the DERIVatives of the system VARIBLs that are associated with "WNCH". Winches are defined as those nodes that control the cable velocity at cable 'ENDS'.
Each WINCH requires 3 state variables:
VARIBL(1)= X position
VARIBL(2)= Y position
VARIBL(3)= Z position
Given the values of the above VARIBLs at the present TIME, as well as the current values of all the other SYSTEM variables, calculate the time DERIVatives of the state variables for a particular winch. The information required to compute these derivatives for a given winch is shared between the PARAM and KONECT arrays. PARAM contains all the physical parameters for the present component, stored in a predetermined order (see MODEND); likewise, KONECT has all the connection data describing links to other SYSTEM components.

SUBROUTINE EQPSET

Process input commands under the SCENE heading that describe the equipment setup.

SUBROUTINE EQPUSE

Process input commands under the SCENE heading that describe the equipment usage.

SUBROUTINE EVENTS

Process input commands under the SCENE heading that describe the special events that occur during a simulation. Currently defined events will reside in /TRNSFR/, /RINGUP/, & /ROLLIN/.

LOGICAL FUNCTION FNDLNK(TYPE,NUMBER,ACTIVE,TYPLFT,NUMLFT,
* TYPRGT,NUMRGT,NAME,IPTLNK)

Search the problem graph for a link that matches all of the specified arguments. If any of these arguments (TYPE through NAME) are passed as '?', treat that specification as a wildcard. Repeated calls to FNDLNK with identical argument lists will cause a search for further matches in the graph, until FNDLNK returns as .FALSE. IPTLNK is a pointer to the link found, if successful; it is not touched otherwise.

LOGICAL FUNCTION FNDNOD(TYPE,NUMBER,ACTIVE,NAME,IPTNOD)

Search the problem graph for a node that matches all of the specified arguments. If any of these arguments (TYPE through NAME) are passed as '?', treat that specification as a wildcard. Repeated calls to FNDNOD with identical argument lists will cause a search for further matches in the graph, until FNDNOD returns as .FALSE. IPTNOD is a pointer to the node found, if successful; it is not touched otherwise.

LOGICAL FUNCTION FNDTAG(TYPE,KK,NN,MM,ITAG)

[illegible] through the list of /TAGS/ to find the one that contains all the

FIGURE 3. SIMUL8 Subroutine Summary
(Continued)

data TYPE, KK,NN,MM. If any of TYPE, KK, NN, or MM are passed as '?', then that field will be treated as a wildcard. In case of wildcards, the first occurance of a satisfactory tag is reported (in ITAG). On subsequant calls with identical arguments, the next occurrences are reported (until no matches are found, when FNDTAG is set .FALSE.)

```
    LOGICAL FUNCTION GETBLK( IBLK,POSITN,NAME)
```

If IBLK is a valid power block serial number, then return the attributes of that component via the arguments. Otherwise, FUNCTION is .FALSE.

```
    LOGICAL FUNCTION GETCBL( ICBL,TENSHN,DRAG,TYPLFT,NUMLFT,TYPRGT,
   * NUMRGT,NAME,IDENTB)
```

If ICBL is a valid cable serial number, then return the attributes of that component via the arguments. Otherwise, FUNCTION is .FALSE.

```
    LOGICAL FUNCTION GETEND( IEND,WEIGHT,VOLUME,NAME,ISEQNC)
```

If IEND is a valid end serial number, then return the attributes of that component via the arguments. Otherwise, FUNCTION is .FALSE.

```
    LOGICAL FUNCTION GETKNK( IKNK,NAME,ISEQNC)
```

If KINK is a valid kink serial number, then return the attributes of that component via the arguments. Otherwise, FUNCTION is .FALSE.

```
    LOGICAL FUNCTION GETKNT( IKNT,WEIGHT,VOLUME)
```

If KNOT is a valid knot serial number, then return the attributes of that component via the arguments. Otherwise, FUNCTION is .FALSE.

```
    SUBROUTINE GETLNK( IPTLNK,TYPE,NUMBER,ACTIVE,TYPLFT,NUMLFT,
   * TYPRGT,NUMRGT,NNAME,NAMES)
```

Retrieve the identifying charactersitics of a LINK in the graph.

```
    SUBROUTINE GETNOD( IPTNOD,TYPE,NUMBER,ACTIVE,NNAME,NAMES)
```

Retrieve the identifying charactersitics of a NODE in the graph.

```
    LOGICAL FUNCTION GETRNG( IRNG,WEIGHT,VOLUME,NAME,ISEQNC)
```

If IRING is a valid ring serial number, then return the attributes of that component via the arguments. Otherwise, FUNCTION is .FALSE.

```
    SUBROUTINE GETTAG( ITAG,TYPE,KK,NN,MM)
```

Report the contents of tag ITAG. The tag TYPE as well as related information KK,NN,MM are returned.

```
    SUBROUTINE GETTBL( TABLE,LMITAB)
```

Read a point pair table from the input stream. This routine acts just like STOTAB, with the addition that a valid dependent table value is the

FIGURE 3. SIMUL8 Subroutine Summary
(Continued)

string 'FREE'. This character flag will be translated into the value -9.999 for storage in the TABLE.

LOGICAL FUNCTION GETTWN(ITWN,TENSHN,DRAG,TYPLFT,NUMLFT,TYPRGT, * NUMRGT,BARLEN)

If ITWIN is a valid twine serial number, then return the attributes of that component via the arguments. Otherwise, FUNCTION is .FALSE.

SUBROUTINE GRAPH

The cable-net problem is generally represented as a "graph" consisting of nodes and links. Use the data in the "basic" common blocks to set up the problem /GRAF/..

SUBROUTINE INPUT

Process all input commands to the SIMUL8 program. In particular, fill the "basic" data commons: /KNOTS/, /RINGS/, /KINKS/, /ENDS/, /TWINE/, /CABLE/, /WINCHS/, /PBLOKS/, /NAME/, /TABLE/, /TAGS/, /SEINER/, and /ENVRON/, /CNTROL/.

SUBROUTINE INTGR8(INDEX)

Integrate the equations of motion for the present system of modules. The GEARS ODE integration package from LLL is featured.

SUBROUTINE KILL(TYPE,NUMBER)

Remove a componant or tag from the problem.

FUNCTION KOMPNT(TYPE)

The value of KOMPNT is the serial number of the mnemonic associated with a particular component TYPE.

FUNCTION LENMOD(TYPE,NUMBER)

Return the length of the module associated with component TYPE,NUMBER. LENMOD is equal to the number of state variables in that module.

FUNCTION LOCMOD(TYPE,NUMBER)

Locate the module of component TYPE, NUMBER. LOCMOD is given the value of the index of the SYSTEM array element that begins the state variables for that component. If the component is not part of the current active graph, return zero.

SUBROUTINE MODBOT(PARAM,KONECT)

The seiner is an active component in the current system. Retrieve its physical and topological characteristics and store them in the PARAM and KONECT arrays, respectively. This constitutes the addition of a new module to the current /STATE/ of the system.

SUBROUTINE MODEND(IEND,PARAM,KONECT)

FIGURE 3. SIMUL8 Subroutine Summary
(Continued)

End number IEND is an active component in the current system. Retrieve its physical and topological characteristics and store them in the PARAM and KONECT arrays, respectively. This constitutes the addition of a new module to the current /STATE/ of the system.

SUBROUTINE MODKNK(IKNK, PARAM, KONECT)

Kink number IKNK is an active component in the current system. Retrieve its physical and topological characteristics and store them in the PARAM and KONECT arrays, respectively. This constitutes the addition of a new module to the current /STATE/ of the system.

SUBROUTINE MODKNT(IKNT, PARAM, KONECT)

KNOT number IKNT is an active component in the current system. Retrieve its physical and topological characteristics and store them in the PARAM and KONECT arrays, respectively. This constitutes the addition of a new module to the current /STATE/ of the system.

SUBROUTINE MODRNG(IRNG, PARAM, KONECT)

RING number IRNG is an active component in the current system. Retrieve its physical and topological characteristics and store them in the PARAM and KONECT arrays, respectively. This constitutes the addition of a new module to the current /STATE/ of the system.

SUBROUTINE MODWCH(IWCH, PARAM, KONECT)

Winch number IWCH is an active component in the current system. Retrieve its physical and topological characteristics and store them in the PARAM and KONECT arrays, respectively. This constitutes the addition of a new module to the current /STATE/ of the system.

SUBROUTINE NET

Read the data from disk that defines the net construction. This net was built during Phase I of the simulation, using the interactive NET program. The data as formatted in the NET file must be converted to the form expressed in the "basic" common blocks. This is hopefully a temporary situation, as the NET program will one day also use the "basic" common data definitions.

LOGICAL FUNCTION NEWTON(YM, XM, YP, XP, XLOWER, XUPPER, ERROR, X)

Perform one step in false positon to estimate the value of X that makes Y(X)=0. Point pairs: YM(XM), YP(XP); bounds on X: XUPPER.GE.X.GE.XLOWER. NEWTON is TRUE if |XP-XM|.LE.ERR.

FUNCTION PACK2(NNNN, MMMM)

The value of PACK2 becomes the concatenation of NNNN to MMMM separated by a decimal point.

COMPLEX FUNCTION PACK4(IIII, KKKK, NNNN, MMMM)

The value of PACK4 is a complex datum containing four integer values packed in the fashion (IIII.KKKK, NNNN.MMMM)

FIGURE 3. SIMUL8 Subroutine Summary
(Continued)

SUBROUTINE PBLOK

Decode the commands, under the SEINER heading, that create power blocks in various positions on board the boat.

FUNCTION PREDCT(HEXT,I,Y,N0)

Extrapolate for time T+HEXT using the Nordsiek array Y for the I'th variable (out of N0 variables).

SUBROUTINE PULL(TIME,INFO,IPOINT,FORCE)

If INFO contains the proper keywords, add tabulated forces to the module that called this routine; i.e. enhance FORCE. IPOINT points into INFO and is bumped as INFO is used.

SUBROUTINE REFRNC(WANT,TYPE,NUMBER)

During processing of an input command, reference is made to a named component. See if the name is recognized by the program. Further, check whether the component is the type that you WANT ('WHCH', 'ENDS', '?'=anything, see COMPNT). If the name passes muster, report the TYPE and serial NUMBER of the component that was referenced.

FUNCTION RNGSUP(AT,CHANGE,H,Y,N0)

This structure change function monitors the slack length of a particular cable. When the cable shortens beyond the specified slack, RNGSUP.GT.0, and a further call to this function with CHANGE=.TRUE. will initiate the structure changes that remove the cable from the problem, transforming rings into knots, and constraining them thereafter.

SUBROUTINE SCENE

Oversee all program input dealing with the SCENE DESCRIPTION.

SUBROUTINE SPARSE(IA,JA,LMTIA,LMTJA)

Create the sparse matrix descriptors, IA & JA, required by the GEARS routine to describe the Jacobian structure. The Jacobian is composed of rectangular blocks of sub-matrices off the main diagonal, and square blocks on the diagonal. Each square block represents a component module in the current system state. The rectangular blocks are situated symetrically w.r.t. the main diagonal, and are associated with active links between various modules. For now, each submatrix is assumed to be full; i.e., the variables within the block are totally interconected.

SUBROUTINE STATES

Use the active nodes and links from the problem /CRAF/ to setup the current /STATE/ of the system. The SYSTEM vector is the collection of all state variables for the problem. Initialize this if necessary. It is sectioned into "modules" consisting of those variables associated with each active node of the graph. Each module is structured according to the requirements of the type of generic component it represents.

FIGURE 3. SIMUL8 Subroutine Summary
(Continued)

All of the parameter data and linking information for separate modules is defined in the /STATE/ common.

PROGRAM SIMUL8

This is the driver for phase II of the FLEXNET package for simulating the geometric response of net-cable systems.

FUNCTION TAGTYP(I)

The value of TAGTYP is the four character mnemonic that identifies TAG type I. The tag types are:

I	MNEMONIC	TAG FORMAT
1	'WGTS'	(1.NODE #, 0.0)
2	'NAMK'	(2.NODE #, NAME #.0)
3	'KINK'	(3.NODE #, NAME #.CABLE SEQ #)
4	'RING'	(4.NODE #, NAME #.CABLE SEQ #)
5	'FLOT'	(5.NODE #, NAME #.FLOAT TABLE #)
6	'ENDC'	(6.NODE #, NAME #.CABLE SEQ #)
7	'ENDF'	(7.NODE #, NAME #.CABLE SEQ #)
8	'CABL'	(8.BAR #, NAME #.CABLE DENSITY TABLE #)
9	'NAMB'	(9.BAR #, NAME #.0)
10	'MODI'	(10.BAR #, 0.0)
11	'PULL'	(11.COMPONENT TYPE, COMP #.TABLE #)
12	'CSTN'	(12.COMPONENT TYPE, COMP #.TABLE #)
13	'PUTN'	(13.COMPONENT TYPE, COMP #.PUTN #)
14	'NAME'	(14.COMPONENT TYPE, COMP #.NAME #)
15	'WNCH'	(15.'WNCH', WINCH #.SPEED TABLE #)
16	'BUOY'	(16.COMPONENT TYPE, COMP #.TABLE #)
17	'PUTC'	(17.COMPONENT TYPE, COMP #.PUTC #)
18	'BOAT'	(18.WINCH #, BLOCK #.0)
19	'CCBL'	(19.'CABL', CABLE#.PUTC#)
20	'WEND'	(20.'ENDS', END#.WINCH#)
21	'INIT'	(21.COMPONENT TYPE, COMP #.IPTSYS)

SUBROUTINE TBLDOT(TABLE,DERIV,LMTTAB)

Given a piece-wise linear TABLE, convert it to a table of DERIVatives of the original values w.r.t. the independent values. Any infinite slopes (derivatives of step functions) will be flagged as errors. The constructed derivative tables will be added onto the end of any tables already in DERIV.

SUBROUTINE UNITV(FROM,TO,UNIT,DISTNC)

Calculate the UNIT vector FROM one point TO another. DISTNC is the actual vector length.

SUBROUTINE UNPAK2(TWO,NNNN,MMMM)

TWO variables contain packed data in the form NNNN.MMMM. This routine extracts the NNNN and MMMM integer halves from the packed word.

SUBROUTINE UNPAK4(FOUR,IIII,KKKK,NNNN,MMMM)

FOUR variables are complex data that contain packed integers in the form (IIII.KKKK,NNNN.MMMM). This routine extracts the four separate integers from the packed record.

FIGURE 3. SIMUL8 Subroutine Summary
(Continued)

FUNCTION VMAG(V)

Calculate the magnitude of a vector

SUBROUTINE WATERV(TIME,POSITN,VWATER)

Provide the water velocity, VWATER, at any POSITN, at TIME.

SUBROUTINE WINCH

Decode the input commands, under the SEINER heading, that create winches at various places on board the boat.

FUNCTION XINTRP(NTAB,X)

Linear interpolation from table NTAB at independent value X.
Outside of table domain, the function value is zero.

SUBROUTINE XYZ(POSITN)

Decode from the input stream a reference to a POSITN in Cartesian space.
Minimum information is sufficient: (X) = (X,0,0)

FIGURE 3. SIMUL8 Subroutine Summary
(Concluded)

/CABLE/

NCABLE	Number of cable segments.
LMTCBL	Limit on storage for cable segments.
CBLTEN(i)	Young's modulus/bulk density for cable segment i.
CBLDRG(3,i)	Lift, minimum, and flow parameters for drag on cable segment i.
CNCTCB(i)	A complex datum, (IIII.JJJJ,MMMM.NNNN), containing the type and serial numbers of the left and right connected components on either end of cable segment i (see COMPNT routine)
KBLNAM(i)	The name serial # of the CABLE to which cable segment i belongs.
KBLDEN(i)	The density table serial # of the CABLE to which cable segment i belongs.

/CNTROL/

TSIMUL	Duration of simulation.
ERROR	Tolerance: The RMS of all state variable absolute errors during any timestep is kept below this
TFRAME	Elapsed time between movie frames.
DTMAX	Maximum timestep allowed.
NATURL	Have the GEARS integrator output frames at the natural system timestep.

/ENDS/

NENDS	Number of ends.
LMTEND	Limit on storage for ends.
WGTEND(i)	The mass of end i.
VOLEND(i)	The volume of end i.
NAMEND(i)	The name serial # of the CABLE to which end i belongs.
ISQEND(i)	The sequence serial # of end i in the CABLE to which it belongs.

/ENVRON/

CURR	The current speed.
DIRCUR	The azimuth to which the current is flowing.
WIND	The wind speed.
DIRWND	The azimuth to which the wind is blowing.

/GRAF/

LENNOD	Number of words used in NODES array.
LMTNOD	Limit on storage for graph nodes.
NODES(i)	Each node requires a part of the NODES array to store several words of information. The contents of this record are described in the ADDNOD routine.
LENLNK	Number of words used in LINKS array.
LMTLNK	Limit on storage for graph links.
LINKS(i)	Each link requires a part of the LINKS array to store several words of information. The contents of this record are described in the ADDLNK routine.

/KINKS/

NKINKS	Number of kinks.
LMTKNK	Limit on storage for kinks.
NAMKNK(i)	The name serial # of the CABLE to which kink i belongs.
ISQKNK(i)	The sequence serial # of kink i in the CABLE to which it belongs.

/KNOTS/

NKNOTS	Number of knots.
LMTKNT	Limit on storage for knots.
WGTKNT(i)	Mass of knot i.
VOLKNT(i)	Volume of knot i.

FIGURE 4. SIMUL8 Data Specifications

/NAME/

NAMES(i)	Logical name identifiers, packed in the "standard fashion".
LMTNAM	Storage limit on array NAMES.

/PBLOCK/

NBLOCK	Number of power blocks.
LMTBLK	Limit on storage for power blocks.
BLKPOS(3,i)	Spatial position of block i relative to the seiner.

/PUTS/

NPUTN	Number of PUT directives.
LMTPUT	Limit on PUT commands.
PUTN(3,i)	Spatial position referenced in PUT directive i.
NPUTC	Number of cable positions to initialize.
LMTPTC	Limit on NPUTC
PUTC(i)	Cable position (alpha) referenced by a 'PUTC' tag.

/RINGS/

NRINGS	Number of rings.
LMTRNG	Limit on storage for rings.
WGTRNG(i)	Mass of ring i.
VOLRNG(i)	Volume of ring i.
NAMRNG(i)	The name serial # of the CABLE to which ring i belongs.
ISQRNG(i)	The sequence serial # of ring i in the CABLE to which it belongs.

/RINGUP/

KBLBGN	Serial number of 'ENDS' that begins the cable being monitored for rings up.
KBLEND	Serial number of 'ENDS' that ends the cable being monitored for rings up.
SLACK	Distance criterion for cable slack that signals rings up condition.

/ROLLIN/

TROLL	Elapsed time after rings up that rollin commences.
LLBLOK	Serial number of 'BLOK' that participates in rollin.
LLSPED	Table number of rollin speed control.

/SEINER/

WINDAF	The projected area of the seiner to the wind, from forward.
WINDAS	The projected area of the seiner to the wind, from the side.
WNDLVR	The wind acting lever arm from the seiner center of gravity.
WETHUL	Seiner hull wetted area.
WATRAF	The projected area of the seiner to the water, from forward.
WATRAS	The projected area of the seiner to the water, from the side.
DRAFT	Seiner draft.
SLNGTH	Seiner length.
SMASS	Seiner mass.
SMOI	Seiner moment of inertia.
SCRWRM	The lever arm from the propeller to the center of gravity.
MOTION(i)	Four motion table numbers: boat speed; angular velocity; thrust; thrust angle.

FIGURE 4. SIMUL8 Data Specifications
(Continued)

/STATE/

NMODUL	Number of modules in the SYSTEM array.
LMTMOD	Limit on storage for modules.
TYPMOD(i)	Character mnemonic of component in module i. (see COMPNT routine)
NUMMOD(i)	Serial # of component in module i.
IPTSYS(i)	Pointer to the start of the SYSTEM variables that define module i.
LMTKCT	Limit on storage for the KONECT array.
LENKNC	Length of KONECT array in use.
KONECT(i)	Linking information for each module in the SYSTEM array.
LMTPAR	Limit on storage for the PARAM array.
LENPAR	Length of PARAM array in use.
PARAM(i)	Physical parameter information for each module in the SYSTEM array.
LMTSYS	Limit on number of state variables in SYSTEM
NEQS	Number of equations to solve in the current system.
SYSTEM(i)	The list of state variables for the current system state, organized by modules.

/TABL/

TABLES(i)	Point pair tables, stacked back to back.
LMTTAB	Storage limit on array TABLES.

/TAGS/

TAG(i)	Complex data packed in the format (IIII.JJJJ,NNNN.MMMM). These data have various meanings (see TAGTYP routine).
NTAGS	Number of tags.
LMTTAG	Storage limit on number of tags.

/TRNSFR/

NXFR	Number of cable end transfers.
LMTXFR	Limit on storage for cable end transfers.
BGNXFR(i)	Beginning time of transfer i.
DURXFR(i)	Duration of transfer i.
IENDXF(i)	Serial number of cable end involved in transfer i.
IWCHXF(i)	Serial number of winch involved in transfer i.

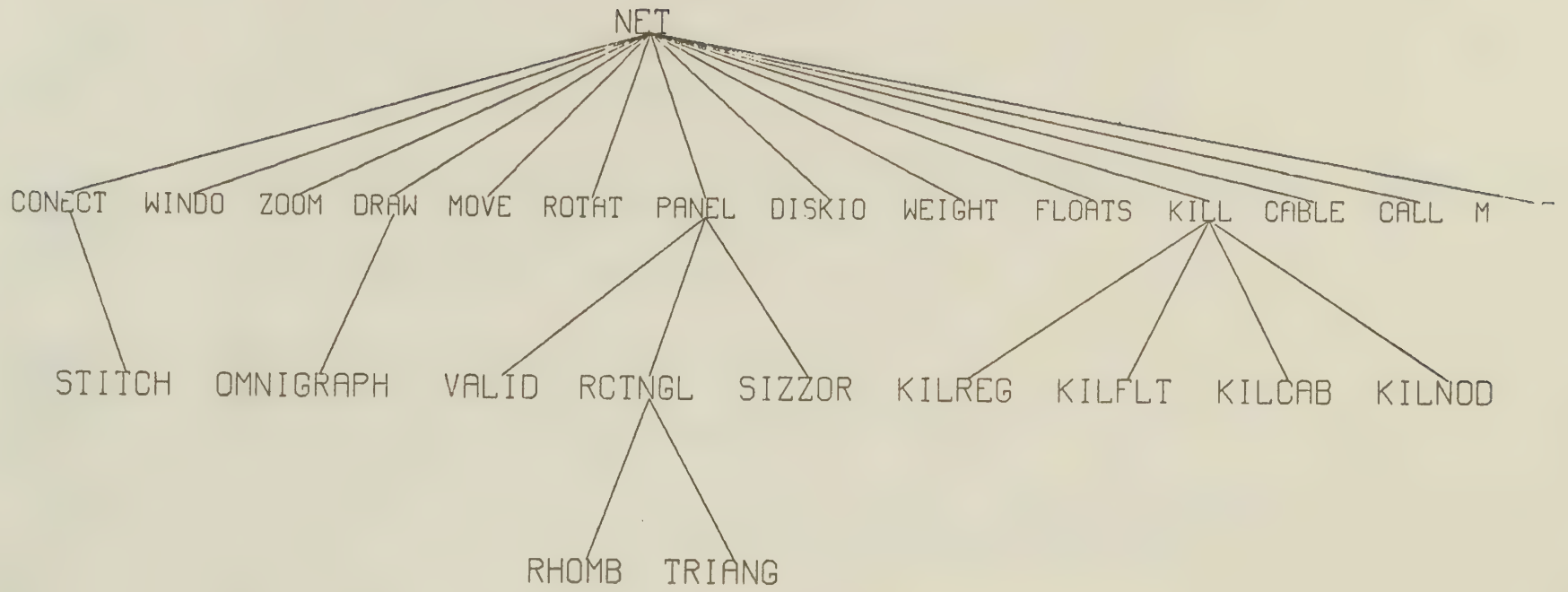
/TWINE/

NTWINE	Number of twines.
LMTTWN	Limit on storage for twines.
TWNLEN(i)	Nominal length of twine i.
TWNTEN(i)	Young's modulus * scaled cross section of twine i.
TWNDRG(3,i)	Lift, minimum, and flow parameters for drag on twine i.
CNCTTW(i)	A complex datum, (IIII.JJJJ,MMMM.NNNN), containing the type and serial numbers of the left and right connected components on either end of twine i (see COMPNT routine)

/WORKNG/

LMTWRK	Limit on size of WORK array required by the GEARS package.
WORK(i)	The working area required by GEARS.

FIGURE 4. SIMUL8 Data Specifications
(Concluded)



UTILITIES

DATA ACCESS/MANAGEMENT

GETKNT, STOKNT, KILKNT, RPLCKT, FETCH
GETBAR, STOBAR, KILBAR, RPLCBR, FNDBAR
GETTAG, STOTAG, KILTAG, FNDTAG, TAGTYP
PACK2, UNPAK2, PACK4, UNPAK4, COLAPS

MISCELLANEOUS

CROSS, INSIDE, INVADE, RECNCT
SMEAR, INPNAM, TRNSFM

FIGURE 5. NET Program Subroutine Hierarchy

SUBROUTINE CABLE(HELP)

Process a cable request: define the statement syntax, derive a table of linear density vs. cable length from a table of cable diameter vs. length; and create new cable segments.

SUBROUTINE CALL(HELP,MODE,NAMREC)

Assign a logical name to a particular 'KNOT' or 'BAR' (MODE); or an alias to some name that exists.

SUBROUTINE COLAPS(MODE,ARRAY,N)

Collapse an ARRAY of N elements. If MODE='FULL', a complete sweep will take place through the array, removing all flaged values. N will be reset accordingly. If MODE='PART', only the last elements of the ARRAY that are flaged in an uninterupted sequence will be removed.
If MODE='TAGS', the complex TAG array will be fully collapsed.

SUBROUTINE CONECT(HELP)

Process the command for connecting one edge of a region to another and then call STITCH.

LOGICAL FUNCTION CROSS(LINE1,L1B,L1E,LINE2,L2B,L2E,T1,T2)

Check whether LINE1, the line segment between points L1B and L1E, intersects LINE2, the segment between L2B and L2E. The endpoints of either line may be composed of 'KNOT' or 'CRNR' points as indicated by LINE1. The intersection parameters are returned in T1 & T2.

SUBROUTINE DISKIO(HELP,MODE)

Save the current state of the net out to the disk file, NET.TMP, if MODE='SAVE'. If MODE='RSTR', restore the net to the condition of the last saved copy. If 'STOR', write a copy of the net to some disk file.

SUBROUTINE FETCH(ITYPE,N,X,Y)

Fetch the coordinates (X,Y) of the N'th 'KNOT', 'CRNR', or 'NEW'.

SUBROUTINE FLOATS(HELP)

Process an ATTACH request: Define the syntax and attach float to knots

LOGICAL FUNCTION FNDBAR(LEFT,IRIGHT,IBAR)

Scan through the list of bar connections to find the one that links LEFT knot to IRIGHT knot. If found, report the serial number of the bar in IBAR. If either LEFT or IRIGHT is specified as '?', then that knot is treated as a wildcard. Repeated calls to FNDBAR with the same input arguments will report subsequant occurrences of matched bars (if any).

LOGICAL FUNCTION FNDTAG(TYPE,KK,NN,MM,ITAG)

FIGURE 6. NET Subroutine Summary

Look through the list of /TAGS/ to find the one that contains all the data TYPE, KK,NN,MM. If any of TYPE, KK, NN, or MM are passed as '?', then that field will be treated as a wildcard. In case of wildcards, the first occurance of a satisfactory tag is reported (in ITAG). On subsequant calls with identical arguments, the next occurrences are reported (until no matches are found, when FNDTAG is set .FALSE.)

SUBROUTINE GETBAR(IBAR,LEFT,IRIGHT,BLEN,BTEN,DRAG1,DRAG2,DRAG3)

Fetch the attributes of bar IBAR.

SUBROUTINE GETKNT(NODE,UMAS,VOLUM)

Fetch the attributes of node NODE.

SUBROUTINE GETTAG(ITAG,TYPE,KK,NN,MM)

Report the contents of tag ITAG. The tag TYPE as well as related information KK,NN,MM are returned. (See TAGTYP)

LOGICAL FUNCTION INPNAM(WANT,NUMBER)

Decipher a reference in the input stream to some 'KNOT' or 'BAR' (depending upon what you WANT). The reference may be made to a serial number or to a logical name associated with the serial #. Logical function fails if the logical name is unknown; if the serial # is out of range; or, if reference is made to a 'BAR' when one wants a 'KNOT' (and vice versa).

LOGICAL FUNCTION INSIDE(ITYPE,IPOINT,IREG)

Check whether point IPOINT is inside region IREG. IPOINT may represent a 'KNOT' or a 'CRNR' according to ITYPE. Points that lie exactly on the boundary are considered outside.

LOGICAL FUNCTION INVADE(LINE,LB,LE,IREG,NCROS,T,EDG)

Determine whether the LINE segment between points LB and LE intersects any of the boundaries of region IREG. The end points of the segment may be 'KNOT' or 'CRNR' points. Parameter values for NCROS intersections along LINE are reported in T(I), along with the edge intersected and the fractional distance packed into EDG(I).

SUBROUTINE KILBAR(NBAR)

Delete bar NBAR from the problem.

SUBROUTINE KILCAB(ICAB)

Delete cable ICAB from the problem.

SUBROUTINE KILFLT(IFLT)

Delete float IFLT from the problem.

FIGURE 6. NET Subroutine Summary
(Continued)

SUBROUTINE KILKNT(N)

Delete knot N and all attached bars.

SUBROUTINE KILL(HELP,NAMES,NAMREG)

Process a KILL request: the doomed item must have a recognized logical name.

SUBROUTINE KILNOD(I)

Delete a bar or node with name number I.

SUBROUTINE KILREG(IREG)

Delete all knots and bars associated with region IREG.

SUBROUTINE KILTAG(ITAG)

Delete tag ITAG from the problem.

SUBROUTINE MODIFY(HELP)

Process a MODIFY request: define the statement syntax, and change the physical properties of bars and knots along a specified existing line.

SUBROUTINE MOVE(HELP)

Translate a region in the X-Y plane.

PROGRAM NET

This is the driver for phase I of the FLEXNET package for modelling net-cable systems. It produces NET data files for use by Phase II, the SIMUL8tion.

FUNCTION PACK2(NNNN,MMMM)

The value of PACK2 becomes the concatenation of NNNN to MMMM separated by a decimal point.

COMPLEX FUNCTION PACK4(IIII,KKKK,NNNN,MMMM)

The value of PACK4 is a complex datum containing four integer values packed in the fashion (IIII.KKKK,NNNN.MMMM)

SUBROUTINE PANEL(HELP)

Define a region of mesh: read in the corners of the region, validate it, calculate boundary lengths, overlay the region with a rectangle of mesh, then SIZZOR the mesh to the region boundary & add selvage.

SUBROUTINE RCTNGL(NCOR,XCORNR,YCORNR)

FIGURE 6. NET Subroutine Summary
(Continued)

Find the smallest rectangle that, when rotated through ANG radians, completely circumscribes the present region. Then generate the proper mesh to fill the rectangle (rectangle may be increased to accomidate an integer number of mesh subunits).

```
SUBROUTINE RECNCT(KNOT1,KNOT2)
```

Transfer all bars from KNOT1 to KNOT2.

```
SUBROUTINE RHOMB(XLEN,YLEN)
```

Generate a regular rhombic mesh to fill a rectangle of dimensions XLEN and YLEN. Increase the size of the rectangle to make room for an integer number of mesh subunits.

```
SUBROUTINE ROTAT(HELP)
```

Rotate a region in the X-Y plane.

```
SUBROUTINE RPLCBR(IBAR,LEFT,IRIGHT,BLEN,BTEN,DRAG1,DRAG2,DRAG3)
```

Replace any or all attributes of bar IBAR. If any variable is passed as 'ASIS', then it will be left as is.

```
SUBROUTINE RPLCKT(NODE,UMAS,VOLUM)
```

Replace any or all attributes of node NODE. If any variable is passed as 'ASIS', then it will be left as is.

```
SUBROUTINE SIZZOR(NREG,NCOR,XCORNR,YCORNR,IEDGE,EDGE,LMTEDG,
* NEWEDG)
```

Given a general polygon region and a rectangle of mesh large enough to cover it, use the region in "cookie cutter" fashion to separate (and eliminate) the meshwork which lies outside of the region from that within. Wherever the region boundary cuts a bar connecting mesh knots, define a new "edge" knot.

```
SUBROUTINE SMEAR(INAM)
```

Render the INAM'th keyword stored in the /NAME/ common unrecognizible.

```
SUBROUTINE STITCH(IEDGE1,IREG1,START1,STOP1,IEDGE2,IREG2,
* START2,STOP2,PCNT,ERSTCH)
```

Connect the mesh along edge IEDGE1 of region IREG1, from START1 meters to STOP1 meters, to edge IEDGE2 of region IREG2, from START2 meters to STOP2 meters.

```
SUBROUTINE STOBAR(LEFT,IRIGHT,BLEN,BTEN,DRAG1,DRAG2,DRAG3,IBAR)
```

Store bar attributes in the first free space available. Return the IBAR serial number.

```
SUBROUTINE STOKNT(UMAS,VOLUM,XKNT,YKNT,IREG,NODE)
```

FIGURE 6. NET Subroutine Summary
(Continued)

Store node attributes in the first free space available. Return the NODE serial number.

SUBROUTINE STOTAG(TYPE,KK,NN,MM)

Find room in the /TAGS/ array to store another TAG variable.

FUNCTION TAGTYP(I)

The value of TAGTYP is the four character mnemonic that identifies TAC type I. The tag types are:

I	MNEMONIC	TAG FORMAT
1	'WGTS'	(1.NODE #, 0.0)
2	'NAMK'	(2.NODE #, NAME #.0)
3	'KINK'	(3.NODE #, NAME #.CABLE SEQ #)
4	'RING'	(4.NODE #, NAME #.CABLE SEQ #)
5	'FLOT'	(5.NODE #, NAME #.FLOAT TABLE #)
6	'ENDC'	(6.NODE #, NAME #.CABLE SEQ #)
7	'ENDF'	(7.NODE #, NAME #.CABLE SEQ #)
8	'CABL'	(8.BAR #, NAME #.CABLE DENSITY TABLE #)
9	'NAMB'	(9.BAR #, NAME #.0)
10	'MODI'	(10.BAR #, 0.0)

SUBROUTINE TRIANG(XLEN,YLEN)

Generate a regular triangle mesh to fill a rectangle of dimensions XLEN and YLEN. Increase the size of the rectangle to make room for an integer number of mesh subunits.

SUBROUTINE TRNSFM(IORDER,X,Y,ANG,X0,Y0)

If IORDER='TRRT', define X,Y relative to new origin X0,Y0, then rotate X,Y ANG radians about X0,Y0. If IORDER='RTTR', rotate X,Y about the current origin, ANG radians, then translate by X0,Y0.

SUBROUTINE UNPAK2(TWO,NNNN,MMMM)

TWO variables contain packed data in the form NNNN.MMMM. This routine extracts the NNNN and MMMM integer halves from the packed word.

SUBROUTINE UNPAK4(FOUR,IIII,KKKK,NNNN,MMMM)

FOUR variables are complex data that contain packed integers in the form (IIII.KKKK,NNNN.MMMM). This routine extracts the four separate integers from the packed record.

LOGICAL FUNCTION VALID(IREG)

Test region IREG to make sure that the first and last corners are not coincident, and the region is not folded over on itself.

SUBROUTINE WEIGHT(HELP)

Process a HANG request: Define the syntax and hang a weight on knots

SUBROUTINE WINDO(HELP)

Accept a new display window size.

SUBROUTINE ZOOM(HELP)

Reset the window display size via the ZOOM command.

FIGURE 6. NET Subroutine Summary
(Concluded)

/BARS/

NBARS	Number of bars
LMTBAR	Storage limit on number of bars.
BAR(i)	Packed data in form NNNN.MMMM denoting left and right connected node serial numbers for bar i.
BARLEN(i)	Fixed length of twine i; if cable segment, the position (ALPHA) of the right connected node.
TENBAR(i)	Young's modulus*scaled cross section for twine; Young's modulus/bulk density for cables.
DRAG(3,i)	lift conponent; minimum; and flow component of drag for bar i.

/CORNRS/

NREG	Number of regions.
LMTREG	Storage limit on number of regions.
NDREGN(i)	Pointers into XCORNR, YCORNR, BONDRY, and IEDGE arrays showing end of data for region i.
XCORNR(i)	X position of corner i.
YCORNR(i)	Y position of corner i.
LMTCOR	Storage limit on number of region corners.
BONDRY(i)	Length of edge i starting at corner i.
EDGE(i)	Packed data (in real format) for each knot in the problem that lies on a region edge. The integer part gives the serial # of the node; the fractional part gives fractional distance of the node along the edge, from the beginning of the edge. These "edge" knots are stored in a specific order: They are sorted first by the edge upon which they lie; and second by the distance from the "start" of that edge.
IEDGE(i)	Pointers into EDGE array: IEDGE(i) indicates the element of EDGE that begins the description of the "edge" knots along edge i.
LMTEDG	Storage limit on number of "edge" knots.
NAMREG(i)	The region names, packed in the "standard fashion".
LMTNRG	Storage limit on NAMREG array.

/NAME/

NAMES(i)	Logical name identifiers, packed in the "standard fashion".
LMTNAM	Storage limit on array NAMES.

/NODES/

NNODES	Number of nodes.
LMTNOD	Storage limit on number of nodes.
UMASS(i)	Unit mass of node i.
VOLUME(i)	Unit volume of node i.
XKNOT(i)	X position of node i.
YKNOT(i)	Y position of node i.
IRECON(i)	Region number, if any, that node i lies within.

/TABL/

TABLES(i)	Point pair tables, stacked back to back.
LMTTAB	Storage limit on array TABLES.

/TAGS/

TAG(i)	Complex data packed in the format (IIII.JJJJ,NNNN.MMMM). These data have various meanings (see TAGTYP routine).
NTAGS	Number of tags.
LMTTAG	Storage limit on number of tags.

FIGURE 7. NET Data Specifications

Section 3

STRUCTURE CHANGE

The modularization of the simulation that was described in Section 2 has greatly facilitated the implementation of various special occurrences that are to be modeled. To illustrate this capability, we have created a "rings up" condition that occurs at some point during the simulation and initiates a structure change. Specifically, when a named cable shortens beyond some point due to winching in, the simulation halts momentarily, removes the cable from the problem, converts the rings that had been sliding along the cable into knots, then starts up again where it left off. This particular structure change involves a change in connectivity as well as the number of equations being solved. Referring back to Figure 2, one can locate the RNGSUP routine that is called by DISCO in the tree diagram. This routine completely defines this structure change; other special events could be inserted in the same way, as they are defined.

To proceed with the illustration, we first create a simple net that contains all the required components that are involved in this structure change (see Figure 8). Next, we define the scenario (see Figure 9). Figure 10 shows the time evolution of this simulation at one second intervals: the net is initialized at t=0. From t=1 to t=5, the winches on either side of the net move independently to the surface. Then the cable is winched in at both ends at 0.1 m/s. Sometime after t=7, less than 1.6 m of cable remains between the winches, and the "rings up" function takes over, performing the structure change. At t=8, it is apparent that the cable is gone (the winches indicated by asterisks remain) and the net begins to sink with gravity. The remainder of the illustrations show the net unfolding and hanging out.

```
"
" Create a very simple net that contains all the features of larger models
" involving netting, rings, and running cables. This net consists of
" a 1m square of mesh with floats on the top edge, and a 2m cable
" running through rings along the bottom edge...
"

WINDO=-1,-1,2,2
PANEL ONE=0,0,1,0,1,1,0,1
C
SCALE=20
C
CALL KNOT 1 (LEFT RING)
CALL KNOT 2 (RIGHT RING)
CALL KNOT 3 (LEFT FLOAT)
CALL KNOT 4 (RIGHT FLOAT)
HANG MASS=.05 ON LEFT RING, RIGHT RING
ATTACH FLOAT (BOB) TO  LEFT FLOAT,RIGHT FLOAT
FLOAT BOB=-100,0,-.1,.001,0,0
CABLE RUN (BIGHT=.5)(THRU  LEFT RING,RIGHT RING)(BIGHT=.5)
CABLE DIAMETER RUN=0,.005
C
CALL KNOT 5 (BEGIN)
CALL KNOT 6 (END)
STORE NET ON FILE:RINGS.NET
STOP
```

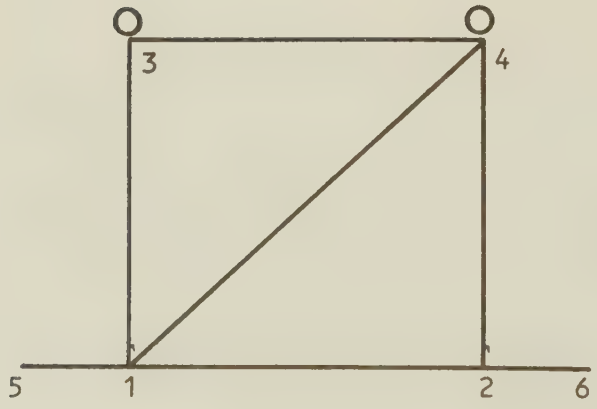


FIGURE 8. RINGS Net: Creation Commands and Schematic

```
"
" This example uses the simple RINGS.NET to illustrate a typical
" structural transition. In particular, the RUNning cable will be
" winched in to a point where it will then be removed from the problem.
" The rings, at that point, will no longer be constrained, and so will
" react to gravity and sink like a weighted net...
"
NET DESCRIPTION
 READ NET FROM FILE:RINGS.NET
SCENE DESCRIPTION
 EQUIPMENT SETUP
  PUT LEFT FLOAT (0)
  PUT RIGHT FLOAT (1)
  PUT LEFT RING (0,0,-.9)
  PUT RIGHT RING (1,0,-.9)
  DEFINE WINCH:BWINCH
  CONNECT BEGIN TO BWINCH
  DEFINE WINCH:EWINCH
  CONNECT END TO EWINCH
  PUT BWINCH (-.3,0,-.9)
  PUT EWINCH (1.3,0,-.9)
 EQUIPMENT USAGE
"
" Both winches will move straight upward. When they reach the surface,
" cable will be winched in at .1m/s from both ends. The floats, meanwhile,
" are fixed to spots on the surface.
"
  MOVE BWINCH TO (-.3,0) OVER TIME=5, AT TIME=0
  MOVE EWINCH TO (1.3,0) OVER TIME=5, AT TIME=0
  SPEED OF BWINCH=0,0, 5,0, 6,-.1, 1M,-.1
  SPEED OF EWINCH=0,0, 5,0, 6,.1, 1M,.1
  CONSTRAIN RIGHT FLOAT DIRECTION(0), SPEED=0,0
  CONSTRAIN LEFT FLOAT DIRECTION(0), SPEED=0,0
 SPECIAL EVENTS
"
" This event defines the nature of the structure change: when the
" amount of cable between the two winches is less than 1.6m, it
" will be removed from the problem...
"
  RINGSUP: RUN CABLE SLACK < 1.6
PROGRAM CONTROLS
 SIMULATION TIME=10
 ERROR CRITERION=.01
 MOVIE FRAME TIME=.5
END
```

FIGURE 9. "Rings Up" Scenario Definition Commands

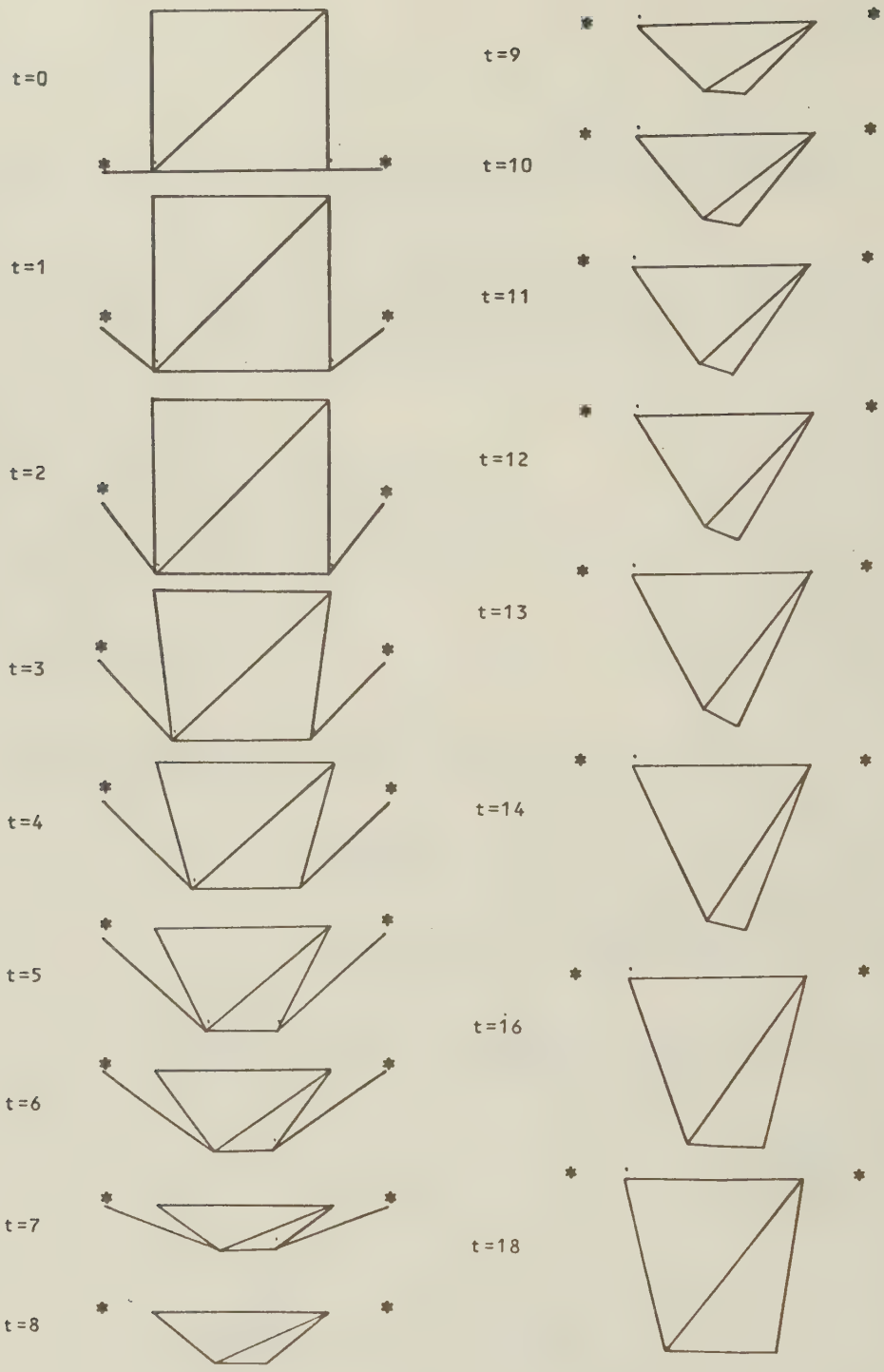


FIGURE 10. "Rings Up" Simulation Time Evolution

Section 4

IMPROVED TREATMENT OF RINGS

The original derivation of the equations of motion of the rings assumed that the cable did not change direction radically on passing through a ring and that the rings would not attempt to pass each other on the cable. The first feature causes a problem because the unit vector along the cable at the ring was required. However, the numerical definition of this was difficult and the method used to define this vector led to unrealistic restrictions on the motion of the cable through the rings. The approach used here is to use the principle of the previous derivation but to demand a reasonable equation of motion for the cable. The two methods coincide for the case where the cable bends smoothly. The second feature is overcome here by retaining information as to the ordering of the rings on the cable and by having them repell each other when they come sufficiently close together with a force similar to the real force between two rings.

4.1 EQUATIONS OF MOTION FOR THE CONSTRAINED CABLE

Figure 11 shows the configuration of interest. Quantities are defined in Tables 1(a) and 1(b). The objective is to find the equation of motion of the ring at $\vec{x}_i$ and the equation of motion of the cable through the ring. One-half of the mass of the cable between two rings and one-half of the force on the cable between two rings is relegated to each of those rings.

The approach used here is to write the momentum equation for the unit consisting of the ring and adjoining net and cables and to write an equation of motion for the cable that is consistent even for the case where the cable bends

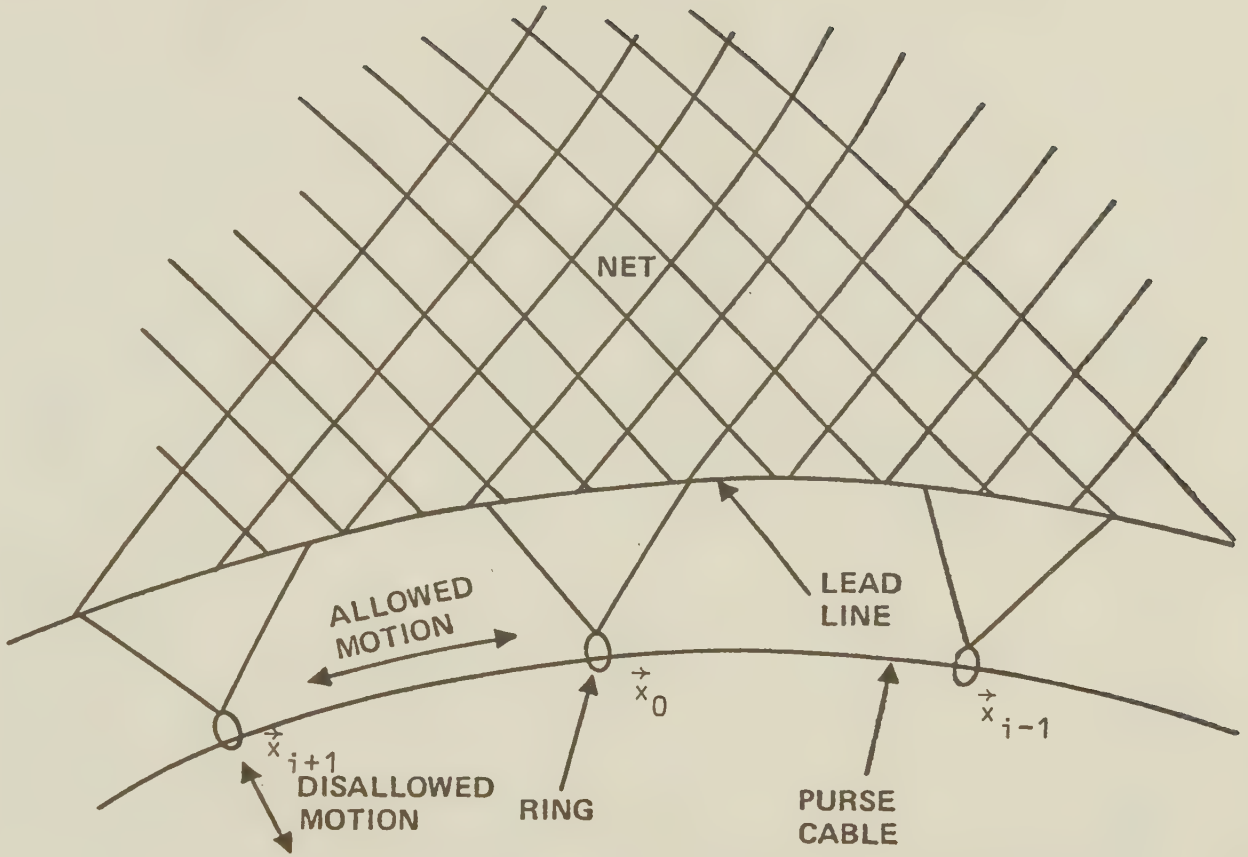


FIGURE 11. Purse Cable and Rings Schematic

Salient characteristics of the purse cable and bow lines are that relative motion through rings is allowed, but relative motion perpendicular to the rings is not allowed; and the cable motion is large compared to ring spacing.

TABLE 1(a). DEFINITIONS USED IN THE EQUATIONS OF MOTION OF THE RINGS

$\vec{x}_j$	SPACIAL POSITION OF j^{th} RING
α_j	POSITION OF j^{th} RING ON CABLE
$\hat{u}_{j+0.5} = \frac{\vec{x}_{j+1} - \vec{x}_j}{\lvert\vec{x}_{j+1} - \vec{x}_j\rvert}$	UNIT VECTOR FROM j^{th} TO $j+1^{th}$ RING
$m_{j+0.5} = \frac{(\alpha_{j+1} - \alpha_j)\ \mu_c}{2}$	MASS OF CABLE AFFECTING j^{th} and $j+1^{th}$ RING
m_j	j^{th} RING MASS
v_j	CABLE SPEED THROUGH: j^{th} RING > $\frac{d\alpha_j}{dt} = - v_j$
$T_{j+0.5}$	TENSION IN CABLE BETWEEN j^{th} and $j+1^{th}$ RING
$\vec{F}_{cj+0.5}$	FORCE ON CABLE BETWEEN j^{th} and $j+1^{th}$ RING (DRAG AND GRAVITY)

TABLE 1. *(Continued)*

TABLE 1(b). ABBREVIATIONS USED FOR i^{th} RING

$\vec{x} = \vec{x}_i$

$\hat{u}_+ = \hat{u}_{i+0.5}$

$\hat{u}_- = \hat{u}_{i-0.5}$

$m_+ = m_{i+0.5}$

$m_- = m_{i-0.5}$

$m = m_i$

$M = m_i + m_+ + m_-$

$v = v_i$

$\vec{F}_+ = T_{i+0.5}\ \hat{u}_+ + \vec{F}_{ci+0.5} / 2$

$\vec{F}_- = T_{i-0.5}\ \hat{u}_- + \vec{F}_{ci-0.5} / 2$

$\vec{F}_N$ = FORCE ON THE NET ATTACHED TO THE i^{th} RING

significantly. These two equations can then be used to determine the equation of motion of the ring alone.

The important quantities are the momentum of a unit

$$\vec{p} = m_+\hat{u}_+v + m_-\hat{u}_-v + M\dot{\vec{x}}$$

with

$$\dot{\vec{p}} = \text{sum of forces} \quad ,$$

where the dot represents time differentiation and the equation of motion of the cable which is of the form:

$$(m_+ + m_-)\dot{v} = -(\dot{m}_+ + \dot{m}_-)v + \tau_+ + \tau_- + \tau_\Delta + f \quad ,$$

where

f is the friction force,

τ_+ is the force from the + cable,

τ_- is the force from the - cable, and

τ_Δ is the force from the net.

Defining the mass averaged vector (which takes place of the unit vector along the cable used in the previous formulation):

$$\bar{u} = \frac{(m_+\hat{u}_+ + m_-\hat{u}_-)}{(m_+ + m_-)} \quad ,$$

the equation of motion of the cable can be substituted into the momentum equation to give:

$$M\ddot{\bar{x}} = \vec{F}_+ + \vec{F}_- + \vec{F}_N - \bar{u}(\tau_+ + \tau_- + \tau_\Delta)$$

$$- v(m_+ + m_-)\,\dot{\bar{u}}$$

The last term can be calculated more easily if it is written:

$$-v(m_+ + m_-)\dot{\bar{u}} = -v\,\frac{d}{dt}\,\bar{u}(m_+ + m_-) + \bar{u}_m\, v\,\frac{d}{dt}\,(m_+ + m_-)\ .$$

It remains to make a few astute choices for the forces on the cable. First, the forces on the cable that contribute to the motion of the cable through the rings is reasonably taken to be the force along the cable:

$$\tau_+ = \hat{u}_+ \cdot \vec{F}_+$$

$$\tau_- = \hat{u}_- \cdot \vec{F}_-$$

The action of the forces on the net is more subtle. Assume that the ring orients itself so that its axis is perpendicular to the cable forces acting on it. This is equivalent to the assumption that the moment of inertia of the ring is small. The component of the force from the net that is along the ring axis will then be the force of importance in moving the ring along the cable. The vector along this direction is:

$$\vec{u}_\perp = \pm\{(\vec{F}_-^2 + \vec{F}_+ \cdot \vec{F}_-)\,\vec{F}_+ - (\vec{F}_+^2 + \vec{F}_+ \cdot \vec{F}_-)\,\vec{F}_-\}$$

with $\hat{u}_\perp = \vec{u}_\perp / |\vec{u}_\perp|$

Since "perpendicular" does not define the sign, the sign of $\hat{u}$ must be chosen to give:

$$\hat{u}_{\perp} \cdot [\vec{x}_{i+1} - \vec{x}_i] > 0$$

in which case,

$$\tau_{\Delta} = - \hat{u}_{\perp} \cdot \vec{F}_N \frac{(m_+ + m_-)}{m} \quad .$$

This leaves the frictional force to be determined. This force is due to components of the force that are not along the ring axis, but along the direction:

$$\vec{u}_F = \vec{F}_+ + \vec{F}_-$$

$$\vec{u}_F = \vec{u}_F / |\vec{u}_F| \quad .$$

The analog with the previous formulation is to take:

$$g = \frac{m(m_+ + m_-)}{M} \left[\frac{\vec{F}_+ + \vec{F}_-}{m_+ + m_-} - \frac{\vec{F}_N}{m} \right] \cdot \hat{u}_F \quad ,$$

and

$$f = - \text{sign } (v) \, \sigma_{ff} \, |g| \quad ,$$

with

$$\sigma_{ff} = \frac{|v| \text{ coefficient of friction}}{\sqrt{v_0^2 + v^2}}$$

where v_0 is some small speed. Again, note that this gives the same equation set as previously when the cable does not bend radically.

It remains to add a force to keep the rings from crossing on the cable. Give the rings an effective thickness, Δ, and assume the normal sequence is such that:

$$\alpha_{i+1} > \alpha_i > \alpha_{i-1} \quad .$$

If

$$\Delta > \alpha_{i+1} - \alpha_i \quad ,$$

a repulsive force should be added between rings i and i+1. This force must act on the rings so that an appropriate force is

$$\delta\vec{F}_N = - \hat{u}_+ \, AY \, \frac{(\alpha_i - \alpha_{i+1} + \Delta)}{\Delta} \, S_+ \quad .$$

where S_+ = sign of $(\alpha_{i+1} - \alpha_i)$ is used to account for the (artificial) switch in the direction of u_+ when the rings are crossed, as happens if the integration takes too large a time step and must recover. Youngs modulus is Y and the contact area is A. The added acceleration on the cable is:

$$\delta[(m_+ + m_-)\,\dot{v}] = AY \, \frac{(\alpha_i - \alpha_{i+1} + \Delta)}{\Delta}$$

A similar situation holds for the case where:

$$\Delta > \alpha_i - \alpha_{i-1} \quad ,$$

when the forces become:

$$\delta\vec{F}_N = S_- \hat{u}_- AY (\alpha_{i-1} - \alpha_i + \Delta)/\Delta \quad ,$$

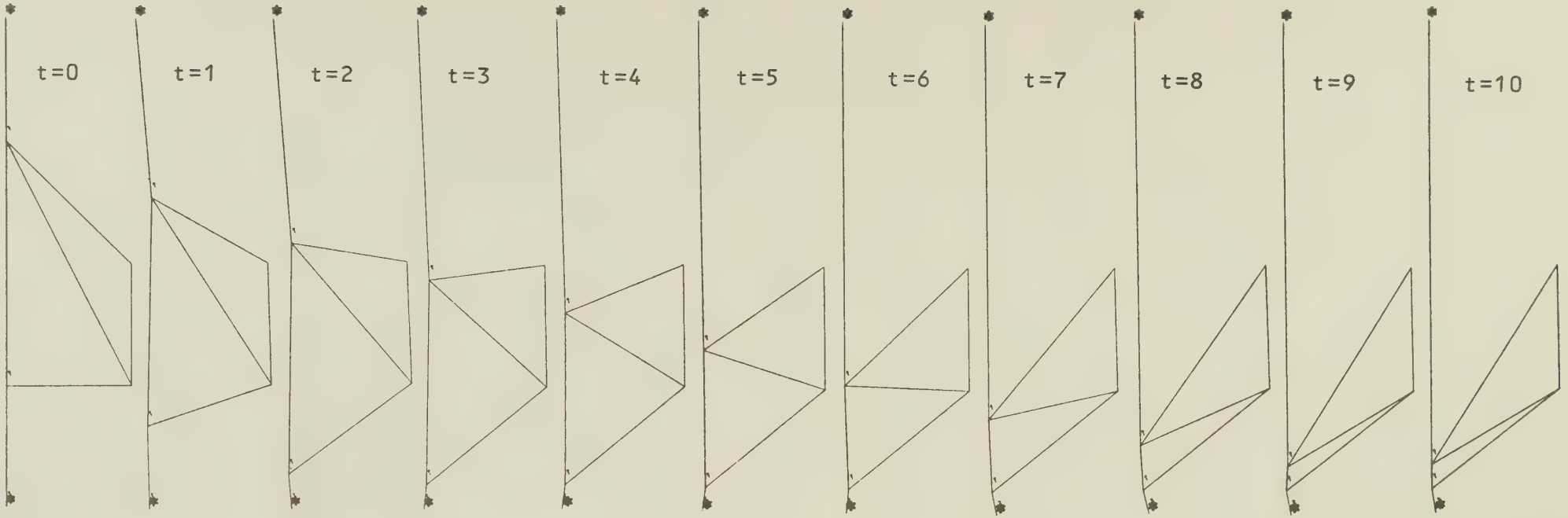
$$S_- = \text{sign of } (\alpha_i - \alpha_{i-1}) \quad ,$$

$$\delta[(m_+ + m_-) \dot{v}] = - AY (\alpha_{i-1} - \alpha_i + \Delta)/\Delta \quad .$$

These forces act abruptly as they would in reality and cause a time step adjustment when the rings meet.

4.2 ILLUSTRATION OF PROPER RING FUNCTIONING

To demonstrate that the rings now act reasonably, we define a scenario that will force the rings against a winch and each other. We make use once again of the RINGS net that was used in the simulation of Section 3. Figure 12 shows the entire simulation. The commands that generate the scenario are listed. The time evolution of the scene proceeds from left to right: the net is initialized at t=0; by t=3, the lower ring has bit the winch; at t=9, the upper ring hits the lower ring and winch. By t=10, the system has come to equilibrium without difficulty.




```
"
" This example uses the simple RINGS.NET to illustrate that the rings
" no longer pose modelling problems. Position the cable vertically and
" drop the rings along the cable till they clank against the bottom winch.
" This cable is barely taut, and will flex sideways somewhat...
"
NET DESCRIPTION
 READ NET FROM FILE:RINGS.NET
SCENE DESCRIPTION
 EQUIPMENT SETUP
  PUT LEFT FLOAT (.5)
  PUT RIGHT FLOAT (.5,0,-.5)
  PUT LEFT RING (0,0,.5)
  PUT RIGHT RING (0,0,-.5)
  DEFINE WINCH NAMED:BWINCH
  CONNECT BEGIN TO BWINCH
  PUT BWINCH (0,0,1)
  DEFINE WINCH NAMED:EWINCH
  CONNECT END TO EWINCH
  PUT EWINCH (0,0,-1)
 EQUIPMENT USAGE
"
" The floats are fixed, one above the other...
"
  CONSTRAIN RIGHT FLOAT DIRECTION(0), SPEED=0,0
  CONSTRAIN LEFT FLOAT DIRECTION(0), SPEED=0,0
PROGRAM CONTROLS
 SIMULATION TIME=10
 ERROR CRITERION=.01
 MOVIE FRAME TIME=.01
 NATURAL TIME
END
```

FIGURE 12. Ring Functioning: Scene Definition and Time Evolution

Section 5

SUMMARY

The principle objectives of the current project were to modularize the code so that the simulation could add or subtract models of components to the system as a function of the configuration and time, and to allow the structure to change as when the oertza is recovered from the skiff. In addition, the modularity was to be such as to permit easy addition of calculations of various quantities of biological significance. Finally, the modeling of the rings was to be fixed to prevent ring crossing.

The primary goals have been achieved.

Secondary, but important goals, have not been achieved. In particular, no large simulation has been run through rings-ups, and rollin has not been attempted. This is a result of the difficulty in constructing the modularized simulation. Because the simulation can dynamically grow, contract, or modify itself, its creation was a significant achievement.

There remain several tasks to be completed before the full simulation can be achieved. In addition to those that remain unfinished after the present project (which amount to the application of the existing modularity to structure changes, and a model for rollin), the main requirements are:

- demonstration on a modern (virtual memory) computer,
- construction of an output processor including movie capabilities for the new data structure,

- modeling of the pre-backdown tie down, and
- full documentation.

The problem of the simulation of the purse seining process is sufficiently well in hand that a set of tasks to complete the simulation can be defined and carried out with a reasonable degree of certainty.